(72) Inventors:
• Cutino, Suzanne C
  Reno, NV 89509 (US)
• Roberts, Shay
  San Franscisco, CA 94118 (US)
• Davis, Virgil M
  Los Altos, CA 92024 (US)

(74) Representative: Finnie, Peter John
Gill Jennings & Every,
Broadgate House,
7 Eldon Street
London EC2M 7LH (GB)

(54) **Cable television payment and load system using smart card**

(57) An architecture and system loads and uses a smart card for payment of goods and/or services purchased on-line over the Internet or directly from a cable provider. A set-top box module on a set-top box controls the interaction with a consumer and interfaces to a card reader which accepts a smart card. Debiting works in conjunction with a merchant server and a merchant payment server. In addition, debiting works in conjunction with a cable head end server and a cable head end payment server. Loading works in conjunction with a bank server and a load server. Alternatively, loading functions in conjunction with a load server when value is loaded from a pre-funded account. To load value, the cable head end server requests a load from a user account at the bank server or a pre-funded account at the load server.
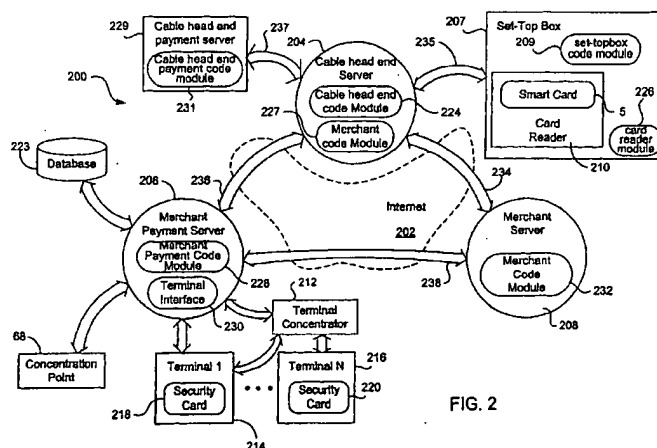
FIG. 2

EP 1 263 230 A1

Description

FIELD OF THE INVENTION

[0001] The present invention relates generally to a payment and value loading system using a computer network. More specifically, the present invention relates to a payment and value loading system for a smart card using an open network such as the Internet and a set-top box.

BACKGROUND OF THE INVENTION

[0002] With the explosive growth in open networks (such as the Internet) over the past several years and the rapid increase in the number of consumers with access to the World Wide Web, there has been a great deal of interest in the development of electronic commerce on the Internet. Traditional financial transactions are being transformed.

[0003] A technique for performing financial transactions uses a smart card. A smart card is typically a credit card-sized plastic card that includes a semiconductor chip for holding the digital equivalent of cash directly, instead of pointing to an account or providing credits. A smart card may include a variety of hardware and software such as a microprocessor, random access memory (RAM), read-only memory (ROM), nonvolatile memory, an encryption module, and a card reader interface that includes the software and hardware necessary for communication with the outside world. However, a smart card may be implemented in many ways, and need not necessarily include a microprocessor or other features. The smart card may be programmed with various types of functionality, such as a stored-value application; credit/debit; loyalty programs, etc. In addition, the card may store a variety of information which might include a card identification number, a personal identification number, authorization levels, cash balances, credit limits, etc.

[0004] FIG. 1 illustrates an environment 100 useful for issuing a smart card (i.e., a smart card including a stored-value application) and reconciling transactions performed with such a card. One possible use of a smart card by a consumer is through the use of a customer operated service payment terminal 50. A customer typically uses such a service payment terminal 50 in a face-to-face environment in order to purchase goods in a store or directly from the terminal itself. A service payment terminal 50 can be an attended device or it can be integrated into a self-service device such as a vending machine or public telephone. For example, the service payment terminal 50 may be incorporated into a soda machine in order to dispense sodas to a customer in which the customer pays by inserting the smart card. Or, the service payment terminal 50 may be a point-of-sale terminal such as is found at a check-out counter where a customer inserts his smart card in order to pur-

chase goods. The user may therefore be provided with a good and/or service once an appropriate value has been deducted from the smart card.

[0005] Such a service payment terminal 50 allows a customer to use a smart card for the payment of goods and/or services, generates a payment result from a transaction, and bundles individual payment results into a collection for transfer to a clearing and administration system, which then transfers funds that had been debited from a customer's smart card to the merchant whose goods and/or services had been purchased from the terminal.

[0006] As shown in FIG. 1, a terminal supplier 102 builds the equipment used by a service provider 104 to provide goods and/or services to customers having a smart card at a service payment terminal 50. Card Supplier 106 contracts with an integrated circuit manufacturer and a card manufacturer for integrated circuits and plastic card bodies, then embeds the integrated circuits into the cards and initializes them with a serial number. It then delivers the cards to card issuer 108. In conjunction with clearing and administration system 110 (such as a system provided by Visa International of Foster City, CA), card issuer 108 personalizes new cards and then transfers these cards to individuals (cardholders 112). The cardholder may then charge the card with value prior to use. Alternatively, the card may come with value already loaded. The cardholder 112 may then use the card at a service payment terminal 50 to purchase goods and/or services from service provider 104. Terminal 50 then debits the value from the card, thus creating a service payment.

[0007] A concentration point is a staging computer that communicates with any number of service payment terminals to collect batches of transactions. The concentration point then sends these transaction batches to a clearing and administration system for processing. Once processed, batch acknowledgments, along with other system updates are sent to the terminals via the concentration point. The concentration point ensures a successful transfer of data between service payment terminals and the clearing and administration system, and prevents overloading of the clearing and administration system. The service provider contracts with a concentration point for collection of the service payments. The concentration point may also be an existing central facility such as a telephone company that collects its own payments from card telephones.

[0008] Periodically, all transactions are sent in a data file from terminal 50 via a concentration point and an acquirer 114 to clearing and batch administration system 110 along with accumulated service payment batches from other terminals. Based upon this collection data, clearing and administration system 110 then receives money from card issuer 108 which had originally come from cardholder 112. Clearing and administration system 110 then transfers a lump sum to acquirer 114 using a suitable settlement service (such as one provided by

Visa International) to pay the various service providers having a relationship with acquirer 114. Based upon the previous collection data, acquirer 114 then transfers an appropriate amount of money to each service provider 104 reflecting the value of the goods and/or services that that service provider had provided that day to cardholders based upon deductions from their smart cards.

[0009]   Thus as described above, a variety of goods or services may be purchased using a smart card from a merchant having a terminal 50, or over the Internet using a technique such as described in U.S. patent application No. 08/951,614 referenced above.

[0010]   However, in order to purchase using a stored value portion of a smart card, the card must be loaded with value first. Value can be loaded onto a smart card in a variety of ways. Currently, it is inconvenient for a user to load value onto his or her smart card. A user must physically travel to a bank or other institution that has an automated teller machine (ATM) or other similar device in order to load value on to his or her smart card. The user can insert money into the machine and have a corresponding value put onto the smart card, the user can use a debit card to deduct value from the user's account at the bank for transfer to the card, or a credit card can be used as the source of funds to be transferred to the smart card. In either case, the user must travel to the bank to load value. Further creating difficulty is that not all banks or other financial institutions have such a machine for loading value onto a user's smart card. One technique for loading value onto a smart card is described in U.S. patent application No. 09/070,488 referenced above.

[0011]   A user encounters similar inconveniences during purchase transactions using a smart card. One example is that of purchase transactions related to goods and/or services made available by a cable provider through a television receiver or set-top box. For instance, when a user wishes to view a pay-per-view movie, the user must typically call a telephone number to provide his or her credit card number. Similarly, when a user wishes to make a purchase (e.g., via the cable provider's shopping channel), the user must typically call a telephone number to provide the necessary information (e.g., card number) to complete the purchase transaction. In either of these examples, the user must use his or her telephone to complete the transaction, preventing the user from using his or her telephone for other purposes during this time period. Thus, the process of completing a purchase transaction using a smart card is time consuming as well as inconvenient. In addition, because the user is billed later for any programming, goods and/ or services from a cable provider, the eventual bill can be extremely high and may be a shock to the user.

[0012]   Even those purchase transactions processed over the Internet can be frustrating to even the most savvy computer user. For instance, each purchase transaction typically requires the entry of a variety of identifying and billing information. At a minimum, such trans-

actions typically require the entry of a card number, name and billing address. As a result, even transactions performed over the Internet can be painfully time consuming and tedious.

[0013]   Accordingly, it would be desirable to have a technique to allow a user to conveniently and easily load value onto a smart card. Moreover, it would be beneficial if a technique were provided to allow a user to conveniently and easily complete a purchase transaction using a smart card. In particular, loading value and purchasing using a smart card in conjunction with a set-top box would be desirable.

SUMMARY OF THE INVENTION

[0014]   In view of the above difficulties a consumer encounters when loading value onto a smart card and completing a purchase transaction using a smart card, the present invention utilizes a set-top box that controls the interaction with a consumer and interfaces to a card reader which accepts the consumer's smart card and allows loading and debiting of the card. For the purposes of this description, a smart card with a stored-value application used in embodiments of the invention will be referred to simply as a "smart card."

[0015]   As described above, a consumer may wish to load value onto a smart card at his or her convenience from home. In addition, a consumer may wish to purchase goods and/or services on-line over the Internet without requiring tedious keyboard entries. Similarly, a consumer may wish to purchase goods and/or services directly from a cable provider without the inconvenience and time-consuming question-and-answer session typically performed over the telephone. To achieve the foregoing, and in accordance with the purpose of the present invention, an architecture and system is disclosed that enables a smart card to be loaded with value on-line over an open network such as the Internet in conjunction with a set-top box. In addition, purchases made over the Internet or directly from a cable provider are similarly made in conjunction with a set-top box.

[0016]   In an embodiment of the present invention, a loading technique allows the consumer to conveniently load value on to his or her smart card from a set-top box via an open network such as the Internet. Through the use of a set-top box, a consumer may connect to his bank or other financial institution. Using appropriate message integrity, value is transferred from the bank to the consumer's smart card. At the same time, the corresponding value is transferred from the bank to the smart card issuer through existing networks for later settlement with a merchant from whom the consumer purchases goods or services. Advantageously, this embodiment makes use of an existing clearing and administration system for eventual settlement of the transaction between the merchant and the card issuer. Also, the transaction is fully auditable and a log of previous transactions is stored on the card for later display. Thus, a

consumer may conveniently load value on to his or her card while a high level of security is maintained and the card issuer can take advantage of unspent funds on the card.

[0017] From the consumer's perspective, the present invention operates in a fashion similar to loading a smart card at an ATM machine, except that the consumer need not insert cash or an additional debit or credit card, nor need travel to a bank. The loading functionality is distributed across the Internet between the card reading device located where the customer is, a bank server holding the consumer's account, and a load server with a host security module that provides security. All of these entities maybe physically remote from one another with router functionality being provided by the Internet.

[0018] From the consumer's perspective, the present invention is easy to use. A consumer need not establish a new relationship with a bank or other Internet service company, nor create a special Internet deposit account in order to load value onto a smart card over the Internet. A consumer simply makes use of his or her bank account and currently available smart cards in order to load value using any conveniently available computing device with a card reader and Internet access.

[0019] In addition, once value has been loaded onto the smart card, a consumer may complete a purchase from a merchant via the Internet or, alternatively, directly from a cable provider. It is important to note that an existing clearing and administration system may be used to reconcile the transaction and to pay the appropriate parties once the value has been spent. Advantageously, a new system and methodology for reconciling transactions need not be developed or implemented. The pre-existing clearing and administration system is used which greatly simplifies implementation of the present invention. A participating bank or holder of a pre-funded account need not implement or become familiar with new procedures for reconciliation of transactions.

[0020] Furthermore, a bank or holder of a pre-funded account need only make a minimal investment in time and money to take advantage of the present invention in order to allow its customers to load value from their existing accounts (e.g., via the Internet). The bank need not engage in the development of complex custom software or accounting procedures. By incorporating software libraries, a bank is ready to begin loading value onto its customer's cards from its web site. Preferably, libraries are provided that interface with an existing server at a bank to facilitate the building of an HTML page. Because a smart card with a stored-value application is used, the bank server, load server and cable head end server perform the details of the transaction and the bank itself is relieved from having to control and keep track of a transaction. Also, the load server and smart card manage and provide security for the transaction. I. e., the bank need not be concerned about security nor be responsible for authenticating a smart card nor for

determining a balance on the card. Of course, a load server could coexist alongside the bank server or could even be the same computer. That is, a bank could implement load server functionality at its own site if it so desired. In a preferred embodiment, the load server and its security module is provided by a separate financial institution or by a third-party processor.

[0021] The present invention also provides benefits to issuers and acquirers. Expansion of the functionality for a smart card increases revenue opportunities from cardholders and merchants. In addition, in one specific embodiment of the invention, funds that are loaded onto a card are transferred from the loading bank to the card issuer so that the issuer may take advantage of the funds on the card until they are spent.

[0022] The present invention is suitable for use with any type of smart card that is able to store an amount and to load a value upon a command. In one embodiment of the invention, a smart card implemented as a processor card works well. Use of a processor card has advantages where information processing is done on the card rather than in the terminal or host computer. Processor cards allow encryption to be done by the card, allow generation of signatures, and can accommodate multiple passwords or personal identification (such as biometrics that uniquely identify the holder of the card). Processor cards also provide increased data security, an anti-fraud capability, flexibility in applications, a multi-purpose capability, and off-line validation. Because high telecommunication costs and/or low reliability of a network may make on-line authorization impractical, a smart card with the capability for performing off-line processing and authentication by itself is extremely valuable.

[0023] In accordance with another aspect of the invention, a consumer may purchase goods and/or services on-line over the Internet or directly from a cable provider. This is accomplished via a set-top box. A consumer uses his or her smart card at a cable head end server via the set-top box in order to purchase goods and/or services from a remote merchant server over the Internet. Alternatively, the consumer may purchase goods and/or services directly from a cable provider via the cable head end server which acts as a merchant server to enable the consumer to make a purchase from the cable provider. The cable head end server sends a draw request to the appropriate payment server. The payment server processes, confirms and replies to the merchant server (optionally by way of the cable head end server).

BRIEF DESCRIPTION OF THE DRAWINGS

[0024] The invention, together with further advantages thereof, may best be understood by reference to the following description taken in conjunction with the accompanying drawings in which:

FIG. 1 is a prior art diagram of an example of a clear-

ing and administration system useful for reconciling financial transactions received from a service payment terminal.

FIG. 2 illustrates an architecture and system for payment over the Internet via a set-top box using a smart card.

FIG. 3 illustrates an architecture and system for loading value onto a smart card via a set-top box.

FIG. 4 illustrates a first payment embodiment of the present invention.

FIG. 5 illustrates an alternative embodiment of the present invention in which the security card releases earlier.

FIG. 6 illustrates yet another alternative embodiment of the present invention having fewer round trip messages between the cable head end server and payment server.

FIG. 7 illustrates still another alternative embodiment of the present invention in which the merchant server compares smart card signatures.

FIG. 8 illustrates an added encryption layer useful for embodiments of the present invention.

FIG. 9 is a flowchart describing a user's perspective of a smart card purchase transaction in accordance with a first payment embodiment of the present invention.

FIGS. 10A-10D are a flowchart describing the processing of a user purchase using the first payment embodiment of the present invention.

FIG. 11 is a flowchart describing the alternative embodiment of FIG. 6.

FIG. 12 is a flowchart describing the alternative embodiment of FIG. 7.

FIG. 13 is a flowchart describing the alternative embodiment of FIG. 8.

FIGS. 14A and 14B are a flowchart describing the added security layer of FIG. 9.

FIGS. 15A-15D are a flowchart describing the loading of value onto a consumer's smart card in accordance with a first load embodiment of the present invention.

FIG. 16 is a flowchart describing a user's perspective of a smart card purchase transaction in accord-

ance with a second payment embodiment of the present invention.

FIGS. 17A-17D are a flowchart describing the loading of value onto a consumer's smart card in accordance with a second load embodiment of the present invention.

FIG. 18 is a block diagram of a typical computer system suitable for use in embodiments of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

GENERAL PAYMENT AND LOAD ARCHITECTURE

[0025]   The present invention separates the functionality involved in a transaction using a smart card in order to take advantage of the routing capabilities of the Internet as well as the capabilities and convenience of a set-top box. FIG. 2 illustrates symbolically an architecture 200 for a payment system involving a smart card A system for loading value onto a smart card is shown in FIG. 3 and may have similar functionality as described below. Shown is an internet 202, a set-top box 207, a cable head end server 204, a cable head end payment server 229, a merchant payment server 206 and a merchant server 208. Local cardholder functions including a consumer card interface, display and accept/cancel options are performed at set-top box 207.

[0026]   A user may wish to purchase goods and/or services from a merchant via the internet or, alternatively, directly from a cable provider. According to a first embodiment enabling purchase transactions to be processed via the internet, payment functions including security card control, data store and use of a concentration point are performed by merchant payment server 206. In addition, the presentation and eventual delivery of goods and/or services by a merchant are performed under control of merchant server 208. The internet 202 performs routing functions between various entities as shown. It should be appreciated that internet 202 may take the form of the Internet currently in use, or may also be any other open network implemented using any combination of computer, telephone, microwave, satellite, and/or cable networks.

[0027]   Basically, set-top box 207 controls the interaction with a user and interfaces to the cable head end server 204, which enables purchase transactions to be processed by either cable head end payment server 229 for purchases made directly from a cable provider or a merchant payment server 206 for purchases made via the Internet. The set-top box 207 includes a card reader 210, a set-top box code module 209, and a card reader module 226. Reader module 226 may be implemented using any suitable software and libraries for communicating with card reader 210 and its actual implementation will depend upon the type of card reader used. Set-

top box code module 209 is responsible for controlling the set-top box as well as controlling displays to the user and for the interaction between the card and the card reader. The module also builds the draw request message after receiving all of the start-up information from the card and the amount of the purchase from the merchant server.

[0028] The card reader 210 accepts a smart card 5 having a stored-value application. Card reader 210 is any suitable interface device that functions to transfer information and commands between cable head end server 204 and smart card 5. By way of example, card reader 210 may be a card reader manufactured by Fischer-Farr International of Naples, Florida, by Hewlett-Packard of Palo Alto, California, by Schlumberger, by Gem Plus, etc. Card reader 210 may take any variety of forms such as a stand alone unit, integrated with the cable head end server, attached to the keyboard of the cable head end server, or even built in to a floppy disk-sized unit capable of being read from a disk drive of the cable head end server, etc.

[0029] According to the first embodiment, merchant payment server 206 communicates directly with a terminal or through a concentrator 212 that handles any number of terminals 214-216 each having a security card 218 and 220 respectively. Merchant payment server 206 also communicates with concentration point 68 for transmission of transaction data to a clearing and administration system. Database 223 stores all suitable information passing through merchant payment server 206 for each transaction. Use of such a database allows any number of merchants (or merchant servers) to use merchant payment server 206 for transactions. Merchant payment server 206 controls payment functions such as handling the attached terminals, managing data base 223 and collection functions. Merchant server 208 is a site that has contracted with an acquirer to accept smart card transactions as payments for goods and/or services purchased over the Internet.

[0030] Smart card 5 may take a variety of forms and is useful in many situations where it is desirable to store monetary value on a card that a consumer may use. In general, a smart card is any card or similar device that is able to store a value that is decremented when the card is used. The card may be purchased complete with a smart or value may be later added to the card by a user. Such cards may also have their value replenished. Of course, a smart card need not be in the form of the traditional credit card, but could appear in any form and of any material that is able to store value and be manipulated by a user for a payment transaction. By way of example, other forms that a smart card may take are any electronic representations. Further, the functionality of smart card 5 may be implemented in software on cable head end server 204, that is, card 5 may be a "virtual" card.

[0031] A smart card may also perform a variety of functions in addition to simply storing value. A card may

be dedicated to the storing value or may contain memory and programs for other applications as well. By way of example, an "electronic wallet" refers to a processor card that can execute a variety of financial transactions and identification functions. Such a card may serve debit, credit, prepayment, and other functions. A smart card typically includes information such as a bank identifier number, a sequence number, a purchase key, a load key, an update key, an expiration date, a transaction counter, a session key, etc., in addition to a running balance.

[0032] A smart card may also be termed a prepayment card, a cash card, or a decrement-in-value card. A smart card may also be implemented by using a variety of card technologies. By way of example, a smart card is typically implemented as a card containing one or more integrated circuits. One example of an integrated circuit card is a memory card that has a semiconductor device for storing information but lacks calculating capability. Another example of an integrated circuit card is a processor card that has not only memory but also a microcontroller to enable the card to make decision. A processor card may also be termed a microprocessor card or a "smart card".

[0033] A processor card may include an encryption module in order to provide a variety of security precautions. By way of example, security precautions may include simple PIN numbers, biometrics, simple algorithms, or sophisticated algorithms such as the Data Encryption Standard (DBS) or Rivest, Shamir, Adelman (RSA) encryption. The card is thus able to use these precautions to verify users, card readers, etc., to validate security cards and/or to provide a unique signature. Preferably card 5 includes any number of keys known to the card issuer that are used during the course of a payment or load transaction to generate signatures for validation of the smart card, validation of the security card or module, and validation of the system itself.

[0034] According to a second embodiment, purchases may be made directly from a cable provider using both a cable head end server 204 and cable head end payment server 229. Cable head end server 204 may be any suitable computer for communicating over a network to a set-top box, a payment server or a merchant server. By way of example, cable head end server 204 may be a mainframe computer, a work station, or a personal computer. Furthermore, cable head end server 204 may also be embodied in any portable device such as a laptop computer, a cellular telephone, or any variety of a personal digital assistant (PDA) such as those made by Apple Computer, Inc. or by U.S. Robotics.

[0035] Cable head end server 204 includes cable head end code module 224. Cable head end module 224 controls communication between the cable head end server, the set-top box and card reader, the payment servers (cable head end payment server 229 and merchant payment server 206) and the merchant server. Cable head end module 224 maybe implemented us-

ing any suitable code. In one embodiment of the invention, cable head end module 224 is implemented using a combination of "C" code and a Java applet. The applet is also supplemented with parameters from an HINT page sent from the merchant server. It is contemplated that Java code works well for implementing the modules on the cable head end, payment and merchant servers because it is platform independent, and could even replace the "C" and "C++" code used. The cable head end module 224 is able to communicate with all components on the Internet, either directly or indirectly.

[0036] The cable head end server 204 is the main channel providing content to the associated set-top boxes. As a central cable facility, it effectively controls at least a portion of the information received by each set-top box to which it connects. Thus, the cable head end server 204 handles messages to and from one or more set-top boxes to which it is connected. The cable head end server therefore performs formatting and routing functions to ensure that the protocol implemented by the cable provider is complied with and the messages are received by the appropriate parties. For instance, since the cable operator may contract with an Internet Service Provider (ESP) to provide it with an Internet connection, the cable head end server 204 may provide translation between the proprietary protocol that the cable company implements and the Internet Protocol. It is also important to note that the cable head end server 204 enables interaction between each set-top box and the appropriate payment server, as will be described below. For instance, the payment server may vary depending upon whether the transaction is performed via the Internet or directly through a server associated with the cable provider.

[0037] In addition, the cable head end server 204 may also include a merchant code module 227, which may be implemented similarly to a merchant code module 232 of a merchant server 208, which will be described in detail below. More particularly, certain transactions may be performed over the internet 202 via an appropriate merchant server 208. For instance, airline tickets, clothing, and other goods may be purchased over the internet 202. However, some goods and/or services may be offered only by the cable provider. As one example, pay-per-view movies are often made available to television viewers by the cable provider. As another example, a shopping channel is typically offered by the cable provider, which enables the television viewer to purchase goods via the cable head end server 204. Merchant code module 227 may therefore process these transactions and may be implemented using any suitable code. By way of example, merchant code module 227 may be implemented using a combination of Perl, HTML, and Java code.

[0038] Payments made in association with transactions processed by the cable head end server 204 may be processed via a dedicated server. More particularly, cable head end payment server 229 includes cable head end payment code module 231. The cable head end payment server 229 processes payment transactions made through the cable head end server 204. As described above, these transactions include, but are not limited to, pay-per-view movie transactions and transactions via the cable provider's shopping channel. In addition, cable head end payment server 229 may be designed to facilitate the communication between the user's smart card and a terminal's security card (not shown to simplify illustration). The communication between the user's smart card and a terminal's security card will be described in further detail below with reference to merchant payment server 206. If a part of a transaction fails to complete, the cable head end payment server 229 may notify the participating system components.

[0039] Traditionally, a cable provider must maintain its own billing system. This requires the employment of billing employees as well as employees to perform other related functions such as customer service. As will be further described with reference to the following figures, through the processing of information obtained from a smart card 5 at a set-top box, the cable provider may transfer the burden of billing onto third parties such as credit card companies. In this manner, money owed to the cable provider may be effectively collected immediately from the credit card companies rather than at the end of a billing cycle.

[0040] Merchant payment server 206 includes merchant payment code module 228 and terminal interface 230. As with cable head end server 204, merchant payment server 206 may be implemented using any suitable computer. By way of example, a personal computer works well. There may be one payment server for each merchant server or a single payment server may service any number of merchant servers. Alternatively, there may be multiple payment servers for a single merchant. In addition, merchant payment server 206 need not be remote from merchant server 208 but may be located at the same site and have a different Internet address, or the payment server and the merchant server may even be implemented on the same computer. Merchant payment server 206 is designed to facilitate the communication between the user's smart card and a terminal's security card. If a part of a transaction fails to complete, the payment server may notify the participating system components.

[0041] Merchant payment module 228 may be implemented using any suitable code. By way of example, merchant payment module 228 is implemented using a combination of "C" code, "C++" code and Java code. Merchant payment module 228 is, in one specific embodiment, a multi-threaded process that can service multiple concurrent cable head end applet transactions on demand. The module is responsible for controlling all interactions with the terminals and their concentrator including the transaction collection function. For individual transactions, the payment module controls the message flows and logs interim results. When an applet con-

nects with the payment server, it creates a transaction thread to support the transaction through its life cycle. Each thread, in turn, assigns a terminal for communication. Having a one-to-one correspondence between transaction threads and terminals has been found to provide desirable results.

[0042] Terminal interface 230 is any suitable set of software and libraries for communicating with a terminal 214 either directly or, as shown, through terminal concentrator 212. The actual implementation of terminal interface 230 will depend upon the type of terminal used. A terminal such as 214 may be any suitable terminal such as are known in the art. By way of example, an iq Delta 2010 terminal made by Schlumberger has been found to provide desirable results. Such a terminal may support a variety of commands originating from the terminal interface. These commands emulate the normal responses that an attached terminal would pass from the smart card to the security card. The actual security card commands are held in the terminal while the terminal performs the tasks necessary to simulate the presence of a smart card.

[0043] Security card 218 may be any suitable security card such as are known in the art (often referred to as a Purchase Secure Application Module--PSAM). In other embodiments, the functionality of security card 218 can be replaced by a hardware security module, could be implemented in hardware within the payment server, or could even be implemented in software.

[0044] By way of example, security card 218 is a removable credit card-sized processor card that is programmed to process and store data relating to financial transactions. Security card 218 contains a microchip embedded in the card that enables the security card to authenticate and to validate the user's smart card. If a user smart card is accepted by the security card, and the smart card contains sufficient value, the security card guarantees that the merchant providing the goods and/or services receives payment according to the amount deducted from the smart card for the goods and/ or services rendered in a preferred embodiment, the security card also contains DES purchase security keys and authenticates the smart card during a purchase transaction and secures the payment and collection totals. A security card also stores signature algorithms for smart cards in use. A security card may also contain a transaction identifier for the current transaction, a financial sum of all transactions remaining to be settled, a session key, and master keys for all smart cards in use. Further, the security card may contain generations of keys, blocked card indicators, date of last update, multiple card programs, different currency rates and additional security.

[0045] Concentration point 68 is a staging computer that communicates with terminals to collect batches of purchase transactions. The concentration point then sends these transaction batches to a clearing and administration system for processing. Once processed, batch acknowledgments, along with other system updates, are sent back to the terminals via the concentration point.

[0046] Merchant server 208 includes a merchant code module 232. Merchant server 208 may be implemented upon any suitable computer capable of communicating with and presenting information to users over an internet. Merchant code module 232 may be implemented using any suitable code. By way of example, merchant module 232 may be implemented using a combination of Perl, HTML, and Java code. Merchant server 208 is typically a generic web server customized for the merchant's business. Merchant server 208 may include databases, CGI scripts and back-office programs that produce HTML pages for an Internet user.

[0047] A brief discussion of the flow of a transaction now follows. In accordance with the first embodiment in which a user performs a purchase transaction via internet 202, during a financial transaction, the cable head end server and merchant server exchange information 234 via internet 202. Each transaction initiated by a user has a transaction identifier created at the merchant server, and a merchant identifier unique to the payment server is also available from the merchant server. Cable head end module 224 and the payment server also use this unique transaction identifier for tracking and logging information about the transaction. Merchant server 208 generates a unique identification of the transaction, completes other required parameters, encrypts as appropriate, and builds an HTML page and sends it to the cable head end server. The set-top box 207 through card reader 210 interacts 235 with the smart card and cable head end module 224 builds a draw request message containing related card information, the purchase amount, and other information supplied by the merchant server or, alternatively, supplied by the merchant code module 227 of the cable head end server 204.

[0048] The cable head end server then communicates 236 with merchant payment server 206, first by forwarding the draw request to the payment server. Merchant payment server 206 verifies the transaction to determine if it is a valid transaction from a known merchant. The transaction is into the payment server's transaction database 223. Upon completion of a transaction, merchant payment server 206 builds a result message containing the identification of the transaction and signs it. The message is then routed to merchant server 208 via cable head end server 204. Merchant server 208 then validates the result message. After determining that the transaction was successful, merchant server 208 creates an HTML page for the purchased information and sends it to cable head end server 204. Alternatively, the merchant may also deliver purchased goods to the user at this point. It is also possible for the payment server and the merchant server to communicate information 238 directly between themselves. Preferably, as cable head end server 204 has already established communication with the merchant server and the payment serv-

er, links 234 and 236 are used to exchange information between the payment server and the merchant server, rather than establishing a new link 238.

[0049] According to the second embodiment, the cable head end server 204 acts as the merchant server and communicates with the cable head end payment server 229. Thus, the cable head end server 204 obtains information associated with the transaction (e.g., from an associated database) when initiated by a user-initiated transaction. Each transaction initiated by a user has a transaction identifier which may be created at the cable head end server 204 or the set-top box 207, and an identifier unique to the cable head end payment server 229 is also available from the cable head end server. Cable head end module 224 and the cable head end payment server also use this unique transaction identifier for tracking and logging information about the transaction. The set-top box 207 through card reader 210 interacts 235 with the smart card and cable head end module 224 builds a draw request message containing related card information, the purchase amount, and other information supplied by the merchant code module 227 of the cable head end server 204.

[0050] The cable head end server then communicates 236 with cable head end payment server 229, first by forwarding the draw request to the cable head end payment server 229. Cable head end payment server 229 verifies the transaction to determine if it is a valid transaction from a known cable provider. The transaction is logged into the payment server's transaction database (not shown to simplify illustration). Upon completion of a transaction, cable head end payment server 229 builds a result message containing the identification of the transaction and signs it. The message is then routed to cable head end server 204. Cable head end server 204 then validates the result message, which maybe provided to set-top box 207. The cable provider may also deliver purchased goods to the user at this point. As shown, the cable head end payment server 229 and the cable head end server 204 may communicate information 237 directly between themselves rather than via the internet 202.

LOADING A SMART CARD

[0051] FIG. 3 illustrates a system 850 for loading value onto a smart card according to one embodiment of the present invention. System 850 includes a cable head end server 204, bank server 860 and load server 862. Cable head end server 204 communicates with card 5 via card reader 210 in set-top box 207, and with bank server 860 and load server 862 over any suitable open network such as Internet 202. Suitable embodiments for the cable head end server, the card reader and the smart card are described above in the description of a payment technique. Preferably, each of cable head end server 204, bank server 860 and load server 862 implement a code module (similar in operation to

the code modules described above) in the Java programming language that provides the functionality described below. For simplicity of explanation, reference will be made below to "cable head end server", "bank server" and "load server" even though the resident code is performing the functions. Card issuer 108 has been described previously in FIG. 1. Card issuer 108 may be a separate financial institution from the bank that includes bank server 860, or card issuer 108 may be the same bank that includes bank server 860.

[0052] According to a first embodiment, the card 5 is loaded through the use of the load server 862 and the bank server 860. Bank server 860 is any suitable computer within a bank or other financial institution. By way of example, bank server 860 is any suitable personal computer, a workstation or a mainframe computer. In one embodiment, bank server 860 runs a "servlet" program (a Java applet running on server) for communication with cable head end 204.

[0053] Load server 862 is also any suitable computer and may be located at a third party location (such as at a processor) or may be located within the same bank as bank server 860. Load server 862 also runs a servlet program for communication with cable head end server 204 and host security module 864. In an alternative embodiment, load server 862 and bank server 860 are the same computer which runs two different applications representing the functionality of each server.

[0054] Host security module (HSM) 864 is a device known in the art that may be embodied in a hardware "black box" or on any suitable computer. The host security module can be implemented in a hardware module outside of load server 862, can be implemented within load server 862, can be implemented in software, or can be implemented as a security card described above. Host security module 864 contains the encryption keys in hardware used for generating signatures (for example S1, S2 and S3) that provide security for the transaction. These signatures are used by smart card 5 and host security module 864 to insure that the card is not expired or counterfeit (i.e., is a valid card), to insure that module 864 is authentic, to insure that system 850 is authentic and, in general, to provide for a valid transaction and to prevent fraud. Card 5 also includes encryption keys for the generation of a smart card signature. In an alternative embodiment, module 864 could be replaced by a standard terminal that includes a security card such as is shown in the previous embodiments. In this situation, the encryption keys would be stored in the security card.

[0055] Briefly, system 850 operates as follows. A consumer accesses bank server 860 via cable head end server 204. Assuming that card 5 is not overloaded and that the user's account with the bank has sufficient funds, the user is able to download value via bank server 860 on to his smart card 5. Cable head end server 204 communicates with load server 862 to receive authorization for the load and for higher security. Card 5 may then be used to make purchases over the Internet as

described earlier in the application or may be used for purchases elsewhere. Once the bank has downloaded value to card 5, a corresponding amount of funds is transferred from the bank to card issuer 108.

[0056]   Card issuer 108 places these funds in a holding pool. Once smart card 5 is used to make a purchase from a merchant, the transaction is captured and settled through a settlement service, such as VisaNet. The issuer bank decrements the funds pool for the amount of the purchase, which is paid to the merchant bank. The merchant bank pays the merchant for the transaction. Settlement may occur in any suitable fashion such as is known in the art and, in particular, may be implemented as previously described in FIG. 1.

[0057]   As described above, the first embodiment enables a bank to download value to smart card 5. However, it maybe desirable to load value to smart card 5 from a pre-funded account rather than from a bank. Thus, according to a second embodiment, the card 5 may be loaded from a pre-funded account. As shown by dotted line 861, value may be loaded directly from load server 862 to cable head end server 204. Alternatively, value may be loaded from load server 862 to cable head end server 204 via the internet 202.

DETAILED PAYMENT ARCHITECTURE

[0058]   FIG. 4 illustrates a detailed embodiment of internet payment architecture 200 having cable head end server 204, merchant payment server 206 and merchant server 208. A smart card 5 is in communication with cable head end server 204, and a security card 218 inside a terminal 214 is in communication with merchant payment server 206. Not shown for simplicity in this figure are other elements of the system shown in FIG. 2. One embodiment of a technique by which a financial transaction may be completed over the Internet will be described in further detail with reference to FIGS. 10A through 10D.

FIRST ALTERNATIVE PAYMENT EMBODIMENT

[0059]   FIG. 5 illustrates an alternative embodiment 200a in which the security card is able to be released sooner than the security card of FIG. 4; this embodiment also requires fewer exchanges between the terminal and the payment server. A security card in a terminal is dedicated to a particular transaction from the moment when the terminal interface selects that terminal until the security card finally issues a "confirmation" message and is released by a terminal interface. Thus, in some circumstances it is desirable to release the security card earlier. By releasing a security card earlier, the card is tied up for a shorter time per transaction and may move on to the next transaction sooner. Also, the less time that a terminal is dedicated to a particular transaction, and the fewer messages exchanged between the two, the less likely chance there is of a communication error

that might interrupt and halt the transaction.

[0060]   Embodiment 200a includes a cable head end server 204, a merchant payment server 206, a merchant server 208, a smart card 5, and a terminal 214 having a security card 218. Communication between the various entities may take place in a similar fashion as in FIG. 4 as indicated by communication links 234, 235, and 236. However, instead of two round trips of information between the terminal and payment server, there is only one round trip in this embodiment.

SECOND ALTERNATIVE PAYMENT EMBODIMENT

[0061]   In another embodiment 200b of the present invention as illustrated in FIG. 6, not only is the security card allowed to release earlier, but the number of messages exchanged between the cable head end server and the payment server are reduced. Instead of comparing smart card signatures in the payment server, the expected smart card signature from the security card is transmitted to the cable head end server where a trusted agent 356 performs the comparison of the expected smart card signature with the actual signature received from smart card 5. Thus, message exchange between the cable head end server and the payment server is reduced to one round trip. This is advantageous in that the time for a transaction is reduced, the security card is released earlier and fewer message exchanges means more reliability over the Internet.

[0062]   Embodiment 200b includes a cable head end server 204, a merchant payment server 206, a merchant server 208, a smart card 5, and a terminal 214 having a security card 218. Communication between the various entities may take place in a similar fashion as in FIG. 4 as indicated by communication links 234 and 235.

THIRD ALTERNATIVE PAYMENT EMBODIMENT

[0063]   FIG. 7 illustrates another embodiment 200c of the invention in which the merchant server performs the comparison of the smart card signature with the expected signature. This embodiment has all of the advantages of the previous embodiment in which the security card is released earlier, and there are also fewer messages passed between the entities. In this embodiment, if the cable head end server is not to be trusted to compare the smart card signatures, then an encrypted signature is passed to the merchant server via the cable head end server. The cable head end server also passes the raw, unencrypted signature from the smart card to the merchant server. A routine 366 in the merchant server then compares the two signatures.

[0064]   Embodiment 200c includes a cable head end server 204, a merchant payment server 206, a merchant server 208, a smart card 5, and a terminal 214 having a security card 218. Communication between the various entities may take place in a similar fashion as in FIG. 4 as indicated by messages 302-306 and communica-

tion link 235.

ENCRYPTION LAYER EMBODIMENT

[0065] FIG. 8 illustrates an embodiment 200d of the present invention in which an encryption layer has been added. Although the present invention may be practiced without this added encryption layer, in a preferred embodiment of the invention, this encryption layer is used. FIG. 8 includes cable head end server 204, merchant payment server 206 and merchant server 208. Other elements of the architecture have been omitted in this figure for simplicity. This extra encryption layer is used not only to protect the contents of messages being transmitted over the Internet, but also to prevent a cable head end server, smart card or other entity from receiving or producing a message that would trick another entity into thinking that a valid transaction had occurred. This encryption also prevents messages from being accidentally or deliberately altered or misdirected.

[0066] It should be appreciated that encryption may be present in any embodiment on all parts of any message sent for security. Preferably, any signature sent over a network is encrypted.

FIRST PAYMENT EMBODIMENT AND
ALTERNATIVES

[0067] As described above with reference to FIG. 2, according to a first payment embodiment, a user may perform a purchase transaction over the Internet via both a merchant server and merchant payment server. In addition, according to a second payment embodiment, a user may perform a purchase transaction via both cable head end server and cable head end payment server for those goods and/or services made available by a cable provider. FIG. 9 is a flowchart describing a method of performing a purchase transaction via the Internet from a user's perspective such as may occur in accordance with the first embodiment of the invention shown in FIG. 2. In step 502, a user acquires and adds value to a smart card. More particularly, value may be added to a stored value portion of a smart card. Alternatively, a user may acquire a smart card that already contains value. This smart card may take the form of any smart card that is able to store value and to debit value from the card. In step 504 the user accesses the merchant server web site via communication link 234 over the Internet via the set-top box and cable head end server. For instance, a user may communicate with the set-top box using a remote control. This access of a web site may be performed in any suitable fashion such as by using any commercially available web browser. In step 506 the user inserts a smart card in card reader 210 at the user's terminal. Alternatively, the user may insert the card before accessing the web site, or even after the selection of goods and/or services from the merchant site. In step 508 the user browses the merchant web site and selects goods and/or services for purchase from the merchant using the web site interface that the merchant has provided. The user then selects an appropriate button on the merchant web site to indicate what the user wishes to purchase. Next, in step 510 the user receives a total sale amount from the merchant server and is directed to actuate a button on the web site indicating that the user wishes to proceed with the purchase using the smart card.

[0068] In step 512 the architecture and system of the present invention (such as is shown in FIG. 2, for example) processes the user order by way of the merchant payment server (via the cable head end server), terminal and security card. In step 514, the user's smart card is debited by the total sale amount and the user receives a "debited" message at the user's terminal. This message is optional if the system is designed so as to not inform the user of this debit. In step 516 the user receives a confirmation message from the merchant server indicating that the transaction has been completed. The user may now download the purchased information and/or receive a receipt for goods and/or services to be rendered or delivered from the merchant at a later date. In step 518 the merchant, via a clearing and administration system, receives payment to its bank account for the goods and/or services rendered by way of information collected from the merchant payment server. In one embodiment of the invention, an existing clearing and administration system is used, as well as an existing methodology for transferring information from a security card for later reconciliation. This use of an existing "back end" allows systems of the invention to be implemented quickly and cheaply. This approach also ensures that cards used in the system are compatible with other smart card terminals.

[0069] One embodiment of a technique by which a financial transaction may be completed over the Internet will now be described using the flowchart of FIGS. 10A through 10D with reference to FIG. 4.

[0070] It should be appreciated that a wide variety of terminology may be used to describe message flow throughout the architecture. For example, the terminology used herein to describe the sequential messages draw request, debit, success, and confirmation, may also be referred to by the respective terminology: draw request, debit IEP, debit response, and debit result (or message result).

[0071] Initially, a suitable web browser of cable head end server 204 is used by the user to access a merchant server web site as indicated by 302. In step 602, the user selects goods and/or services from the merchant site and indicates to the site that the user wishes to purchase these items using a smart card as indicated at 304. In step 604 the merchant server receives this request for a smart card transaction. More particularly, the merchant server receives this request through the cable head end server.

[0072] In step 606 the merchant server builds an

HTML page that includes the following cable head end applet parameters: the total cost of the transaction as determined by the merchant server; the type of currency being used; the port and IP address of the payment server; a unique transaction identifier used by both the payment server and the merchant server to track a transaction; and a unique merchant identifier assigned to the merchant by the acquirer and known to the payment server. Other information may also be included such as the currency's exponent, a status URL address of the merchant server used for communication from the cable head end server, and a merchant server generated key and other security information to ensure the identity of the merchant server and the integrity of the message. Other process related information such as software release level, encryption methodology and keys may also be conveyed. Once this page has been built, the page is sent 306 to the requesting cable head end browser and triggers the loading of the cable head end code module (in this example a Java applet) in the cable head end server.

[0073] Some browsers may not allow an applet to invoke a dynamic link library (DLL) due to security reasons. In an embodiment of the present invention, the cable head end applet along with any DLLs needed are preloaded on the cable head end server. Then, the merchant server is allowed to invoke the cable head end applet and DLLs dynamically to circumvent this security precaution.

[0074] The cable head end server then sends the page to the set-top box in step 607. More particularly, the page may be sent using a protocol proprietary to the cable provider.

[0075] In step 608 the set-top box interacts with smart card 5 in card reader 210 to obtain card information 308 and sends the retrieved information to cable head end server 204. In step 609 cable head end server 204 builds draw request message for later transmission 310 to merchant payment server 206. In one embodiment of the invention, the cable head end applet loads a local DLL, makes an API call to that library, which in turn makes a call to another DLL that finally makes a call to the card reader. In this fashion communication with the card is achieved. Once responses from the card are received, the cable head end module will also combine these responses into a byte stream suitable for transmission over a network to a payment server. Also at this point, the currency type and expiration date on the card are checked, and the total cost of the ordered merchandise is checked against the card balance to ensure that the value on the card is great enough to cover the transaction. If the checks are not successful, a message to that effect is delivered to the user and this transaction terminates.

[0076] The cable head end module emulates a variety of security card commands to receive responses from these commands from the smart card. Because the smart card and the security card are now physically sep-arated from one another, and communication takes place over the Internet, it would not be advantageous to engage in numerous commands and responses over such an open network. In the interest of speed and reliability, it is advantageous to have fewer messages exchanged.

[0077] To operate securely and reliably in this environment, in one embodiment of the present invention, cable head end module 224 emulates a security card and gathers all the responses for transmission in one draw request message. The draw request message may include a variety of information including a draw request token, state information, the merchant identifier, the transaction identifier, security information, a purse provider identifier, an intersector electronic purse (IEP) identifier, an algorithm used by the card, an expiry date, the balance of the card, a currency code, a currency exponent, the authentication mode of the IEP, the transaction number of the IEP, a key version and the purchase amount. As all of this information is prepackaged into a single draw request message, the number of messages between the smart card and the security card over the Internet is greatly reduced.

[0078] In this embodiment, the draw request message is built by packaging the smart card's response to the "reset" and "initialize" commands and any public key certificates along with the total cost and the currency of the transaction received from the HTML page. For public key cards, the card and issuer certificates are obtained from read commands and may also be included in the draw request. By packaging all of this information together into one draw request message, it is possible to cut down on the number of messages exchanged between the cable head end server and the merchant payment server, and reliability and speed is improved. In one embodiment of the invention, an intersector electronic purse (IEP) protocol is used to reset and initialize the card and to receive a response.

[0079] Next, in step 610 the cable head end server accesses the merchant payment server using the IP address received from the merchant server. In step 612 the cable head end server sends the draw request message to the merchant payment server as indicated at 310. The cable head end server also creates a log of this message being sent.

[0080] In step 614 the merchant payment server processes the draw request in conjunction with an associated security card as will be explained in greater detail below with reference to FIG. 10D. Draw request 312 is shown being sent to terminal 214. In one embodiment of the invention, the merchant payment server creates a transaction thread for each connected cable head end module to service it through the life cycle of the transaction. After step 614, the merchant payment server has received a debit command and a security card signature 314 from the security card in the terminal. This debit command may also be termed a "debit IEP" command. The security card signature is a value that uniquely iden-

tifies and validates security card 218 to prove to smart card 5 that the incoming debit command is a valid command from a real security card. This validation ensures that when the smart card is debited, that the financial totals in the security card are updated. Thus, the user of the smart card is guaranteed that a valid debit of the card has occurred. In a preferred embodiment of the invention, the security card signature is an encrypted value ensuring that no other entity can forge an identity of a security card.

[0081]   In step 616 the payment server sends the debit command along with the security card signature to the cable head end server as indicated at 316 for the smart card to debit itself. At this time, the merchant payment server also logs this debit command into its database. The debit command is then sent to the set-top box in step 617.

[0082]   In step 618, upon receiving the debit command from the merchant payment server (via the cable head end server), the set-top box replaces the amount in the debit command with the original amount (from the merchant server) to ensure that the amount has not been tampered with while traveling over the network. At this time, the set-top code module also creates a log of the debit command. Set-top code module 209 then passes the debit command and security card signature to smart card 5 which verifies the signature, debits itself by the purchase amount, and also generates a success message (also termed a "debit response" message) and a smart card signature. The smart card signature is a unique value identifying a valid smart card. In a preferred embodiment of the invention, this signature is in encrypted form to prevent tampering. If card 5 does not have enough value to satisfy the purchase amount, then the "debit response" message indicates as such.

[0083]   In step 620, card 5 sends a success message 320 along with the card signature back to set-top box code module 209 in set-top box 207. This success message may also be termed a "debit response" message. At this point, the purchase amount has been deducted from the balance on smart card 5. In step 621, the success message and card signature are sent to head end server 204 from set-top-box 207. Next, in step 622, cable head end module 224 packages the success message along with the card signature and sends them back to merchant payment server 206 as indicated at 322. Cable head end module 224 also logs the result of this smart card debit.

[0084]   In step 624 the merchant payment server receives incoming message 322 and creates a log and updates the transaction status in its database for future error recovery. The merchant payment server then directs this received message to the security card in the terminal as indicated at 324. Next, in step 626 the security card processes this response from the cable head end's terminal and verifies the received smart card signature.

[0085]   As the security card contains the keys and al-

gorithms necessary to compute smart card signatures, the security card is able to validate that a received smart card signature is in fact a valid one by comparing this smart card signature with a generated expected value. A successful comparison indicates that a success message 324 received from the smart card is in fact a valid success message and that the smart card has been debited. An error result code or a comparison that is not successful potentially indicates that the smart card has not been debited by the proper amount. This comparison of smart card signatures by the security card ensures that a smart card is in fact debited before the merchant server is directed to release the purchased merchandise. This comparison of the smart card signature to an expected value is performed by the security card for the highest level of security. As will be described in the embodiments of FIG. 5, 7, and 8, this comparison of smart card signatures may also take place in the merchant payment server, in the cable head end server or in the merchant server with a variety of other advantages. Assuming that the transaction is so far valid, in step 628 the security card sends a "confirmation" message back to the merchant payment server as indicated at 326. This confirmation message may also be termed a "message result."

[0086]   In step 630 the terminal updates its data store with the smart card number, a transaction count, the total sale amount, the response from the security card, and transaction numbers from the smart card and from the security card. The merchant payment server also logs the response received from the terminal including the merchant identifier, etc., as indicated in step 632. Next, in step 634, the merchant payment server creates a confirmation message including the transaction identifiers and sends this message to the cable head end server in encrypted form as indicated at 328. This message 328 may also be termed a "message result."

[0087]   By sending this confirmation message in encrypted form, the confirmation message maybe passed to the merchant server by way of the cable head end server without fear of tampering. As the confirmation message is encrypted, it would be extremely difficult for the cable head end server or another entity to forge a confirmation message and trick the merchant server into thinking that a transaction had taken place. In another embodiment of the invention, if the cable head end server is a trusted agent, then the confirmation message need not be encrypted. In yet another embodiment, the payment server may sent two confirmation messages, one not encrypted for the cable head end to process, and one encrypted for the merchant server. FIGS. 14A and 14B present an embodiment in which the payment server sends two messages to the cable head end server.

[0088]   At this point, the transaction thread of the payment server that was used for the current transaction may release the terminal, thus allowing the terminal to be used by other transactions. This transaction thread

then exits at this time.

[0089] In step 636 the cable head end server then passes this confirmation message 330 on to the merchant server at the URL address previously received from the merchant server. Message 330 may also be termed a "message result." The cable head end may also post a message to the user informing that the debit has been completed. The cable head end also logs confirmation of the payment. In step 638 the merchant server registers this confirmation message and checks for success. The merchant server calls a validate routine within the merchant code module with the confirmation message in order to validate the response from the cable head end. The validate routine is able to take the transaction identifier along with the encrypted confirmation message to decrypt the confirmation message. If the decrypted confirmation message is acceptable, the merchant server then determines a successful transaction has occurred. Next, in step 640 the merchant server generates an HTML page with the purchase information and delivers this information to the cable head end server. Alternatively, the merchant server may generate a purchase receipt to deliver to the cable head end server indicating goods and/or services to be rendered. At this point, the cable head end server may also log the merchant server's response. The cable head end server then sends the information to the set-top box in step 641. Completion of these steps indicates a successful financial transaction over the Internet using a smart card.

[0090] Returning now to a more detailed discussion of step 614, FIG. 10D describes one technique for processing a draw request message in conjunction with a security card. Once this draw request message has been received by the payment server and passed along to the terminal, the terminal parses the message back into individual responses and passes these responses sequentially to the security card as will be explained below. In an alternative embodiment, a dumb terminal is used and the draw request is parsed into its components and otherwise processed by the payment server, which then sends the responses to the security card itself.

[0091] In step 680 the payment code module of the payment server edits the draw request for syntactic correctness and logs the draw request message as being received. In step 682 the draw request is passed to the terminal interface module of the payment server. In one specific embodiment, the terminal interface then requests a terminal from the payment server's terminal pool. The payment server has a pool of terminals connected to the terminal concentrator that is established at start-up. At start-up, the payment server receives a list of all valid terminal identifiers. The payment server uses these identifiers, and its knowledge of transactions in progress to determine an appropriate terminal to process the transaction. Once a terminal is determined, the terminal interface builds a terminal specific message based upon the draw request and the type of terminal.

[0092] In step 686 the terminal specific draw request

312 is sent to the chosen terminal via the concentrator over a local area network. The concentrator acts as a router between a transaction thread in the payment server and its corresponding terminal. The concentrator looks at a header on the draw request to determine to which terminal the transaction should be routed. In one embodiment of the invention, concentrator 212 is removed and merchant payment server 206 communicates directly with terminal 214 (for example).

[0093] In step 688 the terminal parses the draw request message into its various components and processes each component in turn to emulate a smart card interacting with the security card in a physical terminal. Prepackaging of a variety of information into the draw request message results in fewer exchanges over the Internet between the cable head end server and the payment server. By now simulating an interaction, the security card behaves as if it were in a physical terminal along with the smart card. A variety of responses from a smart card may be emulated. In this embodiment, the terminal sends each of the three packages "answer to reset", "initialize IEP", and "debit" down to the security card individually and waits for a return message before sending the next response. For a public key transaction, the certificates read by the cable head end are also included as individual responses. In this fashion, even though all of the smart card information (the draw request) originating from the cable head end server has been sent at once in prepackaged form over the Internet, the traditional interaction between the smart card and the security card in a physical terminal may be simulated at the terminal in a remote location.

[0094] In step 690 the terminal reaches a "draw amount" state, indicating that the security card is able to generate a debit command. In step 692, the security card generates its security card signature and the debit command. The debit command may also be termed a "debit IEP" command. This signature and debit command 314 are sent to the terminal. The debit command issued by the security card may contain a wide variety of information including the security card identifier, the transaction identifier, the amount to be debited, the currency and currency exponent for the amount, the security card signature, the date, time, and location. The terminal in turn, sends the signature, command, and the terminal identifier to the payment server as indicated in step 694. The information maybe sent to the payment server as indicated at 314 via a concentrator. At this point, step 614 ends and control returns to step 616.

[0095] FIG. 11 is a flowchart that describes a technique for implementing this embodiment with reference to FIG. 5. Step 702 indicates that communication between the various entities takes place in a similar fashion as in FIG. 4 up until the terminal reaches the "draw amount" state. At this point, draw request 312 has been received and processed by the security card. Next, in step 704 the security card generates not only the security card signature and the debit command, but also an

expected smart card signature. This expected smart card signature is a value expected by the security card from the smart card to validate the smart card's success message. This validation will ensure that the smart card has in fact debited itself.

[0096]   In step 706 the security card signature, the debit command and the expected smart card signature are sent to the payment code module in the payment server as indicated at 314a. Also, the terminal updates its data store in a similar fashion as in step 630. Next, step 708 indicates that the transaction occurs as before with reference to step 616-622. The steps indicate that the smart card receives the debit command, debits itself, and returns the success message (also termed a "debit response" message) and its card signature to the payment server.

[0097]   Next, in step 710 the payment server code module processes this response from the smart card by comparing 346 the received card signature with the expected smart card signature received earlier from the security card. This comparison of the two signatures by the payment module of the payment server foregoes the need for another round trip between the payment server and the security card. Because the security card has already delivered the expected card signature to the payment server, the security card may be released as soon as message 314a is received.

[0098]   Assuming that the comparison is successful, the payment module is then able to generate its own confirmation message instead of waiting for a "confirmation" message from the security card. Next, step 712 indicates that the processing continues in a similar fashion as in steps 632-640. The confirmation message is passed on to the merchant server by way of the cable head end server and the merchant server may then deliver the purchased merchandise to the user.

[0099]   FIG. 12 is a flowchart that describes a technique for implementing this embodiment with reference to FIG. 6. Step 722 indicates that communication between the various entities takes place in a similar fashion as in FIG. 4 up until the terminal reaches the "draw amount" state. At this point, draw request 312 has been received and processed by the security card. Next, in step 724 the security card generates not only the security card signature and the debit command, but also an expected smart card signature.

[0100]   In step 726 the security card signature, the debit command and this expected smart card signature are sent to the payment code module in the payment server as indicated in 314a. Also, the terminal updates its data store in a similar fashion as in step 630. Next, in step 728 the payment server code module sends the debit command, merchant signature and expected smart card signature to the cable head end server.

[0101]   Next, step 730 indicates that the transaction occurs as before with reference to steps 618 and 620. The steps indicate that the smart card receives the debit command and debits itself. In step 732, the cable head

end code module itself compares the actual card signature from the smart card with the expected signature from the security card. This comparison of the two signatures by the cable head end module of the cable head end server foregoes the need for another round trip between the payment server and the cable head end server. Also, because the security card has already delivered the expected card signature to the payment server, the security card may be released as soon as message 314a is received.

[0102]   Assuming that the comparison is successful, the cable head end server is then able to generate its own confirmation message in step 734 instead of waiting for a confirmation message from the payment server. Next, step 736 indicates that the processing continues in a similar fashion as in steps 636-640. The confirmation message is passed on to the merchant server and the merchant server may then deliver the purchased merchandise to the user.

[0103]   FIG. 13 is a flowchart that describes a technique for implementing this embodiment with reference to FIG. 7. Step 742 indicates that communication between the various entities takes place in a similar fashion as in FIG. 4 up until the terminal reaches the "draw amount" state. At this point, draw request 312 has been received and processed by the security card. Next, in step 744 the security card generates not only the security card signature and the debit command, but also an expected smart card signature.

[0104]   In step 746 the security card signature, the debit command and this expected smart card signature are sent to the payment code module in the payment server as indicated in 314a. Also, the terminal updates its data store in a similar fashion as in step 630. Next, in step 748 the payment server code module sends the debit command, merchant signature and an encrypted expected smart card signature to the cable head end server. The expected smart card signature is encrypted to prevent tampering by the cable head end server or other outside entity.

[0105]   Next, step 750 indicates that the transaction occurs as before with reference to steps 618 and 620. The steps indicate that the smart card receives the debit command and debits itself. In step 752, the cable head end code module sends the success message, the raw smart card signature and the encrypted signature on to the merchant server. In step 754 the merchant server processes the success message, decrypts the encrypted signature, and compares the two signatures. This comparison of the two signatures by the merchant server foregoes the need for another round trip between the payment server and the cable head end server. Also, because the security card has already delivered the expected card signature to the payment server, the security card may be released as soon as message 314a is received.

[0106]   Assuming that the comparison is successful, the merchant server is then able to generate its own con-

frmation message in step 756 instead of waiting for a confirmation message from the cable head end server. Next, step 758 indicates that the processing continues in a similar fashion as in steps 638 and 640. The merchant server may then deliver the purchased merchandise to the user. In all of the above alternative embodiments, when the transaction is not completed successfully, the payment server reverses the transaction within the terminal.

[0107] FIGs. 14A and 14B are a flowchart describing this embodiment of the invention with reference to FIG. 8. In step 802, the payment server and the merchant server share a unique encryption key. Through a prior business arrangement, both of the servers have arranged to share this unique key to add security to the transaction. The shared key may be of any suitable encryption standard and of any length. The key may be a Data Encryption Standard (DES) key having a length of 128 bits including parity. Although this shared key could be used directly, in a preferred embodiment of the invention, there is a derived unique key for each transaction between the merchant server and the payment server. Alternatively, another encryption standard such as RSA may also be used. Preferably, loading of value is performed under DES, while a purchase may be performed under either DES or public key technology.

[0108] In step 804 the cable head end server and the merchant server engage in a protected Secure Sockets Layer (SSL) session 404 in which a connection is made, a user browses and makes a purchase selection. The SSL session protects the information transmitted over the Internet such as card information, commands, and encryption keys from being discovered by an unauthorized party. Other techniques for protecting a session may also be used.

[0109] In step 806 the merchant server derives a key from the DES key using information unique to the transaction such as the merchant identifier, the transaction identifier, or other information unique to this transaction, such as a random number. Because the payment server shares the DES key with the merchant server and also has access to this unique information about the transaction, the payment server will also be able to derive this same key from the shared DES key. In this step the merchant server also creates a transaction session key (TSK) for use by the cable head end server and payment server in encrypting information.

[0110] In step 808 the merchant server downloads an HTML page of information 406 to the cable head end server that includes the TSK and the TSK that is encrypted using the derived key (ETSK). The TSK encrypted with the derived key will be used by the payment server to return an encrypted (and unreadable by the cable head end) confirmation message to the merchant server. Only the merchant server will be able to decrypt this confirmation message and will thus be guaranteed that a successful transaction has occurred and that merchandise may be released to the cable head end. The

cable head end then sends the page of information to the set-top box at step 809.

[0111] In step 810, the cable head end prepares the draw request in conjunction with the smart card and sends the draw request 408 encrypted with the TSK to the payment server along with the ETSK. In step 812 the payment server uses the shared DES key and the prearranged information unique to the transaction to derive the same key that the merchant server has used. Thus, the derived key can be used to decrypt the ETSK in order to produce the TSK. Once the payment server had produced the TSK, it may decrypt the draw request and process the draw request in any suitable fashion with the security card. Once the payment server has received the debit command from the security card, it encrypts the debit command with the TSK. The debit command may also be termed the "debit IEP command."

[0112] In step 814 the payment server sends the encrypted debit command 410 to the cable head end server. In step 816 the cable head end decrypts the debit command with the TSK it had received earlier from the merchant server and sends the debit command to the set-top box. The set-top box then processes the debit command in a suitable fashion with a smart card in step 817. Once the set-top box has received the debit response message from the smart card, it encrypts this message with the TSK and sends the debit response message 412 to the cable head end server in step 818. The cable head end server then sends the debit response message 412 to the payment server in step 819. In step 820, the payment server decrypts the debit response message with the TSK and processes the debit response message in a suitable fashion with the security card.

[0113] Once the payment server has received a "debit result" message from the security card, the payment server encrypts the "debit result" message with the TSK to form a "debit result C" message for the cable head end. The "debit result C" message will be used by the cable head end server to inform the user of a successful transaction. The payment server also generates its own confirmation message and encrypts the confirmation message with the derived key to form a "debit result M" message. The payment server then sends 414 the "debit result C" message and the "debit result M" message to the cable head end server.

[0114] In step 822 the cable head end server decrypts and processes the "debit result C" message and passes the "debit result M" message 416 on to the merchant server. Because the "debit result M" message is encrypted with the derived key, the cable head end server or other entity is not able to tamper with it. In step 824 the merchant server is able to decrypt the "debit result M" message because it had originally produced the derived key from the DES key. Once the merchant server has determined that a valid "debit result M" message has been received, it confirms that a valid transaction has taken place and may release merchandise to the

user. The merchant server then sends a message indicating success or failure of the transaction to the cable head end server in step 825. The cable head end server then sends the message to the set-top box in step 826.

[0115]   This security embodiment of FIG. 8 may be used with any of the previously described embodiments of the invention. By way of example, this security embodiment may be used with the embodiments of Figures 6 and 7 in which there is only one round trip between the cable head end server and the payment server. In particular, the expected smart card signature received from the security card may be encrypted with the derived key so that it unreadable by the cable head end, yet the merchant server will be able to compare the received smart card signature with the expected card signature to validate the transaction.

[0116]   A wide variety of terminology may be used to describe the keys described above. For example, the keys referred to above as shared DES key, transaction session key (TSK) and derived key, may also be referred to as shared key, session C key and session M key.

LOADING DETAILED TRANSACTION FLOW IN
ACCORDANCE WITH FIRST LOAD EMBODIMENT

[0117]   One embodiment of a technique by which a smart card is loaded over the Internet will now be described using the flowchart of FIGS. 15A through 15D with reference to FIG. 3. Various of the steps below may occur in a different order, the following description is for illustration purposes. Interaction between cable head end server 204 and bank server 860, and between cable head end server 204 and load server 862, is preferably implemented in a similar fashion as between cable head end server 204 and merchant server 208, and between cable head end server 204 and merchant payment server 206 as described above, respectively. Certain implementation details mentioned above with respect to payment are equally applicable to loading a smart card. Furthermore, the exemplary flow shown in the figures illustrates a successful transaction (although a negative result is also explained below in the text). For this reason, a "confirmation" message is referred to, which can more broadly be referred to as a "result" message (to reflect both the possibilities of success and failure of a load). Also, a "load success" message is referred to, which can also be referred to as a "confirmation" message, to reflect its status as either confirming a positive load result or a negative load result.

[0118]   Initially, a suitable web browser of cable head end server 204 is used by the user to access a bank server Internet site. In step 871 the user selects an option to load value onto card 5 (via the cable head end server). In step 872 the bank server sends a request for card information (including current card balance and maximum card balance); set-top box 207 reads the current card balance, currency, and other card information via card reader 210 and returns the balance to bank

server 860 via the cable head end server 204. In step 873 the bank server determines the maximum load value and verifies that enough funds are in the user's account to accommodate a load request.

[0119]   In step 874 the bank server builds an HTML page that includes the following cable head end applet parameters: the maximum load value; the type of currency being used; the port and IP address of the load server; a unique transaction identifier used by both the load server and the bank server to track a transaction; a unique bank identifier assigned to the bank and known to the load server; and a session key. Other information may also be included such as the currency's exponent, a status URL address of the bank server used for communication from the cable head end server, and other security information to ensure the identity of the bank server and the integrity of the message. Other process related information such as software release level, encryption methodology and keys may also be conveyed. Once this page has been built, the page is sent to the requesting cable head end browser and triggers the activation of the cable head end code module (in this example a Java applet) in the cable head end server. The page is then sent to the set-top box in step 888 so that the user may confirm the load value which is to be loaded onto the smart card.

[0120]   To determine the load value, the bank server requests that the user enter the amount to load to the card. Assuming that the user's account is adequate, the bank server requests the user's account be debited in step 875 by the load value. Advantageously, the debit request from the bank server can use the existing ATM and accounting systems of the bank to debit the user's account. From the bank's point of view, value is being transferred from the user's account much in the same way that value would be transferred to a user in the form of cash at an ATM. In this situation, though, the value is not being dispensed as cash at an ATM, but is being sent over the Internet to a smart card.

[0121]   In step 876 the set-top box interacts with smart card 5 in the card reader to obtain card information and sends the card information to the head end server. In step 889 the head end server builds a load request message for later transmission to load server 862. Once responses from the card are received, the cable head end server combines these responses into a byte stream suitable for transmission over a network to a load server.

[0122]   The cable head end server emulates a variety of host security module 864 commands to receive responses from these commands from the smart card. The smart card and the security module are physically separated from one another; communication takes place over the Internet. In the interest of speed and reliability, it is advantageous to have only the traditional authentication, response, and confirmation messages exchanged.

[0123]   To operate securely and reliably in this environment, in one embodiment of the present invention

the cable head end server emulates a security module and gathers all the responses for transmission into one load request message. The load request message may include a variety of information and preferably includes a first card signature (termed S1), a card number, an expiry date, and a load amount. Other information such as the security algorithm, transaction counter, current card balance, and bank server time stamp are also preferably provided.

[0124] As all of this information is prepackaged into a single load request message, the number of messages exchanged between the smart card and the security module over the Internet is minimized.

[0125] Next, in step 877 the cable head end server accesses the load server using the IP address received from the bank server. In step 878 the cable head end server sends the load request message to the load server. In step 879 the load server processes the load request in conjunction with an associated host security module 864 as will be explained in greater detail below with reference to FIG. 15D. After step 879, the load server has received an issuer security module signature (termed S2) as part of a load command from the security module 864. The security module signature is a value that uniquely identifies and validates the security module to prove to smart card 5 that the incoming load command is a valid command from a real security module. Thus, the user of the smart card, and other interested parties are guaranteed that a valid load of the card has occurred. In a preferred embodiment of the invention, the security module signature is an encrypted value ensuring that no other entity can forge an identity of a security module.

[0126] In step 880 the load server sends the load command including with the security module signature to the cable head end server. In step 893 the cable head end server then sends the load command to the set-top box for the smart card to load itself. In step 881, upon receiving the load command from the load server, the set-top box passes the load command to smart card 5 which verifies the signature, loads itself by the load value, and also generates a load success message, a second smart card signature (termed S3), and a result code indicating success or failure of the load. In a preferred embodiment of the invention, this signature is in encrypted form to prevent tampering.

[0127] In step 882, card 5 sends load success message containing the card signature (S3) and result code back to the set-top box. The set-top box then sends the success message containing the card signature (S3) and result code to cable head end server in step 894. Next, in step 883 cable head end server 204 packages the load success message along with the card signature and sends them back to load server 862. In step 884 the load server receives the incoming message. The load server then processes the message into its components and directs the components to the security module. Next, in step 885 the security module may process this

response from the cable head end's terminal and verify the received smart card signature (S3).

[0128] As the security module contains the keys and algorithms necessary to compute smart card signatures, the security module is able to validate that a received smart card signature is in fact a valid one by comparing the received smart card signature with a generated expected value. A successful comparison indicates that a load success message received from the smart card is in fact a valid success message and that the smart card has been loaded. Assuming that the transaction is so far valid, in step 886 the security module sends a "confirmation" message back to the load server.

[0129] It is possible that the smart card has not been loaded by the proper amount, that the card is invalid, a user is fraudulent or another discrepancy. For example, it is possible that a user has tampered with the card to make it appear that a load has not occurred, when in fact a load has occurred. In this situation, processing in step 882 and on is slightly different. For example, instead of generating a "load success" message, the card my generate a "negative result" code, potentially indicating that the card has not been loaded. Processing of this situation would then occur as follows.

[0130] In step 882, card 5 sends a load message containing the result code and smart card signature S3 back to cable head end server 204. Cable head end server 204 recognizes a negative result code, and invokes negative result handling. Cable head end server 204 interacts with card 5 and generates a new load request for a zero value load using elements from the original request, along with a new card signature S1.

[0131] The negative result code, along with the signatures S3 and new S1, and the zero value load request are passed to the load server for analysis. The load server determines if the transaction counter in the zero value load equals the transaction counter in the previous request, along with verifying other pertinent information such as date and time, card number, and currency code and exponent. If the transaction counters are the same, then it is possible that a valid negative result has been received, but it should be verified because the cable head end is not trusted. If the counters are equal, the load server will hold the original S3 and will generate a new load request to the security module using data element values that would have been expected if the original transaction had failed. The new load request along with the new S1 is sent to the security module. The security module then compares the original S1 (from the original load request) to the new S1. If S1 is valid, then the original negative result is true and the security module generates a signature to confirm to the load server that there was no load. The original negative result from the card is then released to the security module to complete the original transaction. Processing would continue, but a user account would not be debited, and no settlement need occur because the card was, in fact, not loaded. If S1 is not valid, the negative response is not

true and then the result code in the original request is changed to reflect a successful load and passed to the security module. Processing then continues reflecting that a load has occurred.

[0132]   On the other hand, if the transaction counters are not the same, then it is still possible that a valid negative result has been received, but it should be verified because the cable head end is not trusted. First, the load server decreases the transaction counter in the new load request to match that of the original. The request along with the new S1 is passed to the security module. The security module calculates its own new S1 based upon the modified new load request. If there is no match, it means that the negative result was in error and that the card had been loaded. Processing continues to reflect a loaded card. If there is a match, it means the negative result was correct and that the transaction counter had been increased by accident. The user account is not debited, and not settlement occurs.

[0133]   Returning now to further processing, in step 887 the load server logs the response received from the security module and updates its database with the transaction identifier, the bank identifier, the load value, etc. In general, any of the plethora of information passing through the load server maybe added to its database. Next, in step 890 the load server creates a confirmation message including the transaction identifier and sends this message to the cable head end server in encrypted form. By sending this confirmation message in encrypted form, the confirmation message maybe forwarded to the bank server by way of the cable head end server without fear of tampering. As the confirmation message is encrypted, it would be difficult for the cable head end server or another entity to forge a confirmation message and trick the bank server into thinking that a valid load had taken place.

[0134]   In step 891 the cable head end server forwards the confirmation message on to the bank server at the URL address previously received from the bank server. The cable head end server may also post a message to the user informing that the load has been completed. The cable head end server may also log confirmation of the load. In step 892 the bank server registers the confirmation message. The bank server calls a routine to decrypt the confirmation message. If the decrypted confirmation message is acceptable, the bank server determines a successful load has occurred. The confirmation message provides assurance to the bank that the user's card was in fact loaded with a particular value and prevents fraud. For example, a fraudulent user who tries to claim that his bank account was decremented and his card not loaded (and should thus receive more money from the bank) would be thwarted because the confirmation message proves that the user's card was in fact loaded. Alternatively, the "confirmation" message may indicate that a load did not occur, in which case the account would not be debited, and no settlement would occur.

[0135]   At this point a successful load of the user's card has occurred (assuming all is well). For example, if the user had requested $100, that amount has been decremented from the user's account at the bank, and $100 has been loaded onto the user's smart card. Preferably, at this point the amount loaded (in this example $100) is transferred from the bank to the smart card issuer preferably through an existing network. The $100 is transferred so that the card issuer may manage the float on these unspent funds until the user spends the $100. Once the $100 (or a smaller portion) has been spent with a merchant, the card issuer is then able to settle the transaction with the merchant using any suitable clearing and administration system. In alternative embodiment, the bank may retain the $100 and settle directly with the merchant. In another embodiment, the bank and the card issuer are the same financial institution, and the $100 may be shifted between parts of the organization or remain in place.

[0136]   Returning now to a more detailed discussion of step 879, FIG. 10D describes a technique for processing a load request message in conjunction with a security module. Once the load request message is received by the load server, the load server parses it into the appropriate elements and passes a request to the security module as will be explained below. Alternatively, the load server can build a network message and switch the request to a remote authentication server. Or, a smart terminal could parse the message and pass responses to the security module.

[0137]   In step 895 the load server edits the load request for syntactic correctness and logs the request as received. In step 896 the load server constructs a load request message. In step 897 the load server passes the load request to the security module to emulate a smart card interacting with the security module. The load server behaves as if a smart card were actually interacting in an ATM (for example) through a network to a host with a security module. In this fashion, the load request originating from the cable head end server has been sent in prepackaged form over the Internet emulating a traditional interaction between the smart card in an ATM.

[0138]   In step 898, the security module verifies the received smart card signature (S1) to prevent fraud. The security module generates its security module signature (termed S2) and the load command. The signature S2 will confirm to the cable head end server and the smart card that the host security module is authentic and belongs to the issuer of the smart card. Additionally, S2 protects against a user trying to perform a fake load, keys out of synchronization, a counterfeit card, an expired card, etc. The security module then sends the signature and load command to the load server as indicated in step 899. At this point, step 879 ends and control returns to step 880.

SECOND PAYMENT EMBODIMENT

[0139]   Rather than purchasing goods and/or services from a merchant via the Internet, a user may wish to purchase goods and/or services directly from a cable provider. FIG. 16 is a flowchart describing a method of performing a payment transaction via cable head end server and cable head end payment server from a user's perspective such as may occur with the second alternative of the embodiment of the invention shown in FIG. 2. In step 1102, a user acquires and adds value to a smart card. Alternatively, a user may acquire a smart card that already contains value. This smart card may take the form of any of the above-described smart cards that are able to store value and to debit value from the card. In step 1104 the user accesses the cable head end server which, as described above, operates as a merchant server for the cable provider. In step 1106 the user inserts a smart card in card reader 210 at the set-top box. Alternatively, the user may insert the card before accessing the cable head end server, or even after the selection of goods and/or services from the cable head end server. In step 1108 the user selects goods and/or services for purchase from the cable provider using the interface that is provided by the cable head end server. As one example, the user may purchase goods and/or services from a shopping channel made available by the cable provider. As another example, the user may wish to view a pay-per-view movie and indicate this selection using the interface. Next, in step 1110 the user receives a total sale amount from the cable head end server and is directed to actuate a button indicating that the user wishes to proceed with the purchase using the smart card.

[0140]   In step 1112 the architecture and system of the present invention (such as is shown in FIG. 2, for example) processes the user order by way of the cable head end payment server, terminal and security card. In step 1114, the user's smart card is debited by the total sale amount and the user receives a "debited" message from the cable head end module. This message is optional if the system is designed so as to not inform the user of this debit. In step 1116 the user receives a confirmation message from the cable head end server indicating that the transaction has been completed. The user may now download the purchase information and/or receive a receipt for goods and/or services to be rendered or delivered from the cable head end server, operating in this instance as the merchant server for the cable provider. In step 1118 the cable provider, via a clearing and administration system, receives payment to its bank account for the goods and/or services rendered by way of information collected from the cable head end payment server. In one embodiment of the invention, an existing clearing and administration system is used, as well as an existing methodology for transferring information from a security card for later reconciliation. This use of an existing "back end" allows systems of the invention

to be implemented quickly and cheaply. This approach also ensures that cards used in the system are compatible with other smart terminals. A purchase transaction may be processed similarly to the process shown and described with reference to FIGS. 10A-10D above.

LOADING DETAILED TRANSACTION FLOW IN ACCORDANCE WITH SECOND LOAD EMBODIMENT

[0141]   As described above with reference to FIGS. 15A through 15D, a smart card may be loaded from a bank account via a bank server. However, it may be desirable to load value onto a smart card from a pre-funded account that is independent from any bank account. One embodiment of a technique by which a smart card is loaded directly via the load server will now be described using the flowchart of FIGS. 17A through 17D with reference to FIG. 3. Various of the steps below may occur in a different order; the following description is for illustration purposes. Interaction between cable head end server 204 and load server 1062 is preferably implemented in a similar fashion as between cable head end server 204 and merchant server 208, and between cable head end server 204 and merchant payment server 206 as described above, respectively. Certain implementation details mentioned above with respect to payment are equally applicable to loading a smart card. Furthermore, the exemplary flow shown in the figures illustrates a successful transaction (although a negative result is also explained below in the text). For this reason, a "confirmation" message is referred to, which can more broadly be referred to as a "result" message (to reflect both the possibilities of success and failure of a load). Also, a "load success" message is referred to, which can also be referred to as a "confirmation" message, to reflect its status as either confirming a positive load result or a negative load result.

[0142]   Initially, a suitable web browser of cable head end server 204 is used by the user to access a load server Internet site or, alternatively, a load server may be accessed directly as indicated at 861 of FIG. 3 via cable head end server 204. In step 1071 the user selects an option to load value onto card 5. In step 1072 the load server sends a request for card information (including current card balance and maximum card balance); set-top box 207 reads the current card balance, currency, and other card information via card reader 210 and returns the balance to load server. In step 1073 the load server determines the maximum load value and verifies that enough funds are in the pre-funded account to accommodate a load request.

[0143]   Assuming communication with the load server takes place via the Internet, in step 1074 the load server builds an HTML page that includes the following cable head end applet parameters: the maximum load value; the type of currency being used; the port and IP address of the load server; a unique transaction identifier used by both the load server to track a transaction; a unique

identifier assigned to the pre-funded account and known to the load server; and a session key. Other information may also be included such as the currency's exponent, a status URL address of the load server used for communication from the cable head end server, and other security information to ensure the identity of the load server and the integrity of the message. Other process related information such as software release level, encryption methodology and keys may also be conveyed. Once this page has been built, the page is sent to the requesting cable head end browser and triggers the activation of the cable head end code module (in this example a Java applet) in the cable head end server. The cable head end server then sends the page to the set-top box in step 1088 in order to request that the user confirm the amount to load to the card. Assuming that the pre-funded account is adequate, the load server requests the pre-funded account be debited in step 1075 by the load value.

[0144] In step 1076 the set-top box interacts with smart card 5 to obtain card information and sends the card information to the cable head end server. In step 1089 the cable head end server builds a load request message from the card information for later transmission to load server 1062. Once responses from the card are received, the cable head end server combines these responses into a byte stream suitable for transmission over a network to a load server.

[0145] The cable head end server emulates a variety of host security module 1064 commands to receive responses from these commands from the smart card. The smart card and the security module are physically separated from one another; communication may take place over the Internet. In the interest of speed and reliability, it is advantageous to have only the traditional authentication, response, and confirmation messages exchanged.

[0146] To operate securely and reliably in this environment, in one embodiment of the present invention the cable head end server emulates a security module and gathers all the responses for transmission into one load request message. The load request message may include a variety of information and preferably includes a first card signature (termed S1), a card number, an expiration date, and a load amount. Other information such as the security algorithm, transaction counter, current card balance, and load server time stamp are also preferably provided.

[0147] As all of this information is prepackaged into a single load request message, the number of messages exchanged between the smart card and the security module is minimized.

[0148] Next, in step 1077 the cable head end server accesses the load server using the IP address received from the load server. In step 1078 the cable head end server sends the load request message to the load server. In step 1079 the load server processes the load request in conjunction with an associated host security

module 1064 as will be explained in greater detail below with reference to FIG. 15D. After step 1079, the load server has received an issuer security module signature (termed S2) as part of a load command from the security module 1064. The security module signature is a value that uniquely identifies and validates the security module to prove to smart card 5 that the incoming load command is a valid command from a real security module. Thus, the user of the smart card, and other interested parties are guaranteed that a valid load of the card has occurred, In a preferred embodiment of the invention, the security module signature is an encrypted value ensuring that no other entity can forge an identity of a security module.

[0149] In step 1080 the load server sends the load command with the security module signature to the cable head end server. The cable head end server then sends the load command with the security module signature in step 1093 to the set-top box for the smart card to load itself. In step 1081, upon receiving the load command from the load server, the set-top box passes the load command to smart card 5 which verifies the signature, loads itself by the load value, and also generates a load success message, a second smart card signature (termed S3), and a result code indicating success or failure of the load. In a preferred embodiment of the invention, this signature is in encrypted form to prevent tampering.

[0150] In step 1082, card 5 sends load success message containing the card signature (S3) and result code back to the set-top box. The set-top box then sends the load success message containing the card signature (S3) and result code to cable head end server 204 in step 1094. Next, in step 1083 cable head end server 204 packages the load success message along with the card signature and sends them back to load server 1062. In step 1084 the load server receives the incoming message. The load server then processes the message into its components and directs the components to the security module. Next, in step 1085 the security module may process this response from the cable head end's terminal and verify the received smart card signature (S3).

[0151] As the security module contains the keys and algorithms necessary to compute smart card signatures, the security module is able to validate that a received smart card signature is in fact a valid one by comparing the received smart card signature with a generated expected value. A successful comparison indicates that a load success message received from the smart card is in fact a valid success message and that the smart card has been loaded. Assuming that the transaction is so far valid, in step 1086 the security module sends a "confirmation" message back to the load server.

[0152] It is possible that the smart card has not been loaded by the proper amount, that the card is invalid, a user is fraudulent or another discrepancy. For example, it is possible that a user has tampered with the card to

make it appear that a load has not occurred, when in fact a load has occurred. In this situation, processing in step 1082 and on is slightly different. For example, instead of generating a "load success" message, the card my generate a "negative result" code, potentially indicating that the card has not been loaded. Processing of this situation would then occur as follows.

[0153] In step 1082, card 5 sends a load message containing the result code and smart card signature S3 back to cable head end server 204. Cable head end server 204 recognizes a negative result code, and invokes negative result handling. Cable head end server 204 interacts with card 5 and generates a new load request for a zero value load using elements from the original request, along with a new card signature S1.

[0154] The negative result code, along with the signatures S3 and new S1, and the zero value load request are passed to the load server for analysis. The load server determines if the transaction counter in the zero value load equals the transaction counter in the previous request, along with verifying other pertinent information such as date and time, card number, and currency code and exponent. If the transaction counters are the same, then it is possible that a valid negative result has been received, but it should be verified because the cable head end is not trusted. If the counters are equal, the load server will hold the original S3 and will generate a new load request to the security module using data element values that would have been expected if the original transaction had failed. The new load request along with the new S1 is sent to the security module. The security module then compares the original S1 (from the original load request) to the new S1. If S1 is valid, then the original negative result is true and the security module generates a signature to confirm to the load server that there was no load. The original negative result from the card is then released to the security module to complete the original transaction. Processing would continue, but a user account would not be debited, and no settlement need occur because the card was, in fact, not loaded. If S1 is not valid, the negative response is not true and then the result code in the original request is changed to reflect a successful load and passed to the security module. Processing then continues reflecting that a load has occurred.

[0155] On the other hand, if the transaction counters are not the same, then it is still possible that a valid negative result has been received, but it should be verified because the cable head end is not trusted. First, the load server decreases the transaction counter in the new load request to match that of the original. The request along with the new S 1 is passed to the security module. The security module calculates its own new S1 based upon the modified new load request. If there is no match, it means that the negative result was in error and that the card had been loaded. Processing continues to reflect a loaded card. If there is a match, it means the negative result was correct and that the transaction counter

had been increased by accident. The user account is not debited, and not settlement occurs.

[0156] Returning now to further processing, in step 1087 the load server logs the response received from the security module and updates its database with the transaction identifier, the account identifier, the load value, etc. In general, any of the plethora of information passing through the load server may be added to its database. Next, in step 1090 the load server creates a confirmation message including the transaction identifier and sends this message to the cable head end server in encrypted form. By sending this confirmation message in encrypted form, the confirmation message may be forwarded without fear of tampering. As the confirmation message is encrypted, it would be difficult for the cable head end server or another entity to forge a confirmation message and trick the load server into thinking that a valid load had taken place.

[0157] In step 1091 the cable head end server may forward the confirmation message on to the user via the set-top box. In this manner, the cable head end server may post a message to the user informing that the load has been completed. In addition, the cable head end server may forward the confirmation message on to another server at a URL address previously received from the load server. The cable head end server may also log confirmation of the load. Once the additional server receives the confirmation message, it may call a routine to decrypt the confirmation message. If the decrypted confirmation message is acceptable, the additional server determines a successful load has occurred. The confirmation message provides assurance that the user's card was in fact loaded with a particular value and prevents fraud. For example, a fraudulent user who tries to claim that his account was decremented and his card not loaded (and should thus receive more money from the pre-funded account) would be thwarted because the confirmation message proves that the user's card was in fact loaded. Alternatively, the "confirmation" message may indicate that a load did not occur, in which case the account would not be debited, and no settlement would occur.

[0158] Returning now to a more detailed discussion of step 1079, FIG. 10D describes a technique for processing a load request message in conjunction with a security module. Once the load request message is received by the load server, the load server parses it into the appropriate elements and passes a request to the security module as will be explained below. Alternatively, the load server can build a network message and switch the request to a remote authentication server. Or, a smart terminal could parse the message and pass responses to the security module.

[0159] In step 1095 the load server edits the load request for syntactic correctness and logs the request as received. In step 1096 the load server constructs a load request message. In step 1097 the load server passes the load request to the security module to emulate a

smart card interacting with the security module. The load server behaves as if a smart card were actually interacting in an ATM (for example) through a network to a host with a security module. In this fashion, the load request originating from the cable head end server has been sent in prepackaged form (e.g., over the Internet) emulating a traditional interaction between the smart card in an ATM.

[0160] In step 1098, the security module verifies the received smart card signature (S1) to prevent fraud. The security module generates its security module signature (termed S2) and the load command. The signature S2 will confirm to the cable head end server and the smart card that the host security module is authentic and belongs to the issuer of the smart card. Additionally, S2 protects against a user trying to perform a fake load, keys out of synchronization, a counterfeit card, an expired card, etc. The security module then sends the signature and load command to the load server as indicated in step 1099. At this point, step 1079 ends and control returns to step 1080.

COMPUTER SYSTEM EMBODIMENT

[0161] FIG. 18 illustrates a computer system 900 suitable for implementing an embodiment of the present invention. Computer system 900 includes any number of processors 902 (also referred to as central processing units, or CPUs) that are coupled to storage devices including primary storage 906 (such as random access memory, or RAM) and primary storage 904 (such as a read only memory, or ROM). As is well known in the art, primary storage 904 acts to transfer data and instructions uni-directionally to the CPU and primary storage 906 is used typically to transfer data and instructions in a bi-directional manner. Both of these primary storage devices may include any suitable of the computer-readable media described below. A mass storage device 908 is also coupled bi-directionally to CPU 902 and provides additional data storage capacity and may also include any of the computer-readable media described below. Mass storage device 908 may be used to store programs, data and the like and is typically a secondary storage medium (such as a hard disk) that is slower than primary storage. It will be appreciated that the information retained within mass storage device 908, may, in appropriate cases, be incorporated in standard fashion as part of primary storage 906 as virtual memory. A specific mass storage device such as a CD-ROM 914 passes data uni-directionally to the CPU.

[0162] CPU 902 is also coupled to an interface 910 that includes one or more input/output devices such as such as video monitors, track balls, mice, keyboards, microphones, touch-sensitive displays, transducer card readers, magnetic or paper tape readers, tablets, styluses, voice or handwriting recognizers, biometrics readers, or other computers. CPU 902 optionally may be coupled to another computer or telecommunications

network using a network connection as shown generally at 912. With such a network connection, it is contemplated that the CPU might receive information from the network, or might output information to the network in the course of performing the above-described method steps. Furthermore, method embodiments of the present invention may execute solely upon CPU 902 or may execute over a network connection such as the Internet in conjunction with a remote CPU that shares a portion of the processing.

[0163] In addition, embodiments of the present invention further relate to computer storage products with a computer readable medium that have program code thereon for performing various computer-implemented operations. The media and program code may be those specially designed and constructed for the purposes of the present invention, or they may be of the kind well known and available to those having skill in the computer software arts. Examples of computer-readable media include, but are not limited to: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROM disks; magneto-optical media such as floptical disks; and hardware devices that are specially configured to store and execute program code, such as application-specific integrated circuits (ASICs), programmable logic devices (PLDs) and ROM and RAM devices. Examples of program code include machine code, such as produced by a compiler, and files containing higher level code that are executed by a computer using an interpreter.

[0164] Although the foregoing invention has been described in some detail for purposes of clarity of understanding, it will be apparent that certain changes and modifications may be practiced within the scope of the appended claims. For instance, any suitable smart card capable of loading, storing and decrementing value on command may be used with the present invention. Also, any network capable of performing routing functionality between a cable head end server and a load and bank server may be used. Furthermore, the security module may be a physically separate module, a card located in a terminal attached to a load server, or its functionality may be incorporated directly into a load server in hardware or software. And although the cable head end server may be used to route messages between the bank server and load server, both of these servers may also communicate directly between themselves, and may even be the same computer. The specific messages shown passing between the computers are exemplary, and other types of messages may be used. A specified load request is shown, but other information may also be loaded onto a smart card using a security module emulation and then sent packaged as one message to the security module over a network. In addition to monetary value, other types of value such as electronic cash, checks, awards, loyalty points, benefits, etc., may be loaded onto a card, and the term "value" is intended to broadly cover all these various types. Any suitable type

of encryption may be used to encrypt messages passing between the computers. Therefore, the described embodiments should be taken as illustrative and not restrictive, and the invention should not be limited to the details given herein but should be defined by the following claims and their full scope of equivalents.

Claims

1. A network payment system for transacting a sale over a network using a smart card, said network payment system comprising:

> a cable head end server in communication with said network;
> a set-top box in communication with said cable head end server, said set-top box including a card reader for communicating with a smart card;
> a merchant server in communication with said network, said merchant server having at least an item of merchandise or service for sale; and
> a payment server in communication with said network, said payment server including an interface for communicating with a security card and being arranged to receive a purchase message including an indication of said purchase transaction and to transmit a confirmation message to said merchant server over said network, whereby said merchant server is authorized to release said item of merchandise or said service to a user associated with said smart card.

2. The network payment system according to claim 1, wherein said network is an internet and said merchant server includes a merchant web site for advertising said item of merchandise or service for sale over said internet.

3. The network payment system according to claim 1 or 2, wherein each of said cable head end server, said merchant server and said payment server are at a separate location and communicate over said internet.

4. The network payment system according to any preceding claim, further comprising:

> a clearing and administration system for reconciling a plurality of transactions over said network.

5. The network payment system according to any preceding claim, wherein said cable head end server further includes a command emulator for emulating security card commands that are sent to said smart card and for grouping responses to said security card commands into a draw request message to be sent to said payment server, and said payment server includes a response emulator for emulating responses from said smart card that are sent to said security card.

6. The network payment system according to any preceding claim, wherein said payment server includes a comparator for comparing a smart card signature received from said smart card with an expected signature received from said security card to confirm a transaction.

7. The network payment system according to any preceding claim, wherein said merchant server includes a comparator for comparing a smart card signature received from said smart card with an expected signature from said security card received via said payment server, whereby a transaction is confirmed from said payment server.

8. A cable television payment system for transacting a sale directly from a cable provider using a smart card, said cable television payment system comprising:

> a cable head end server in communication with said network, said cable head end server acting as a merchant server having at least an item of merchandise or service for sale by a cable provider;
> a set-top box in communication with said cable head end server, said set-top box including a card reader for communicating with a smart card; and
> a payment server in communication with said cable head end server, said payment server including an interface for communicating with a security card and being arranged to receive a purchase message including an indication of said purchase transaction and to transmit a confirmation message to said merchant server over said network, whereby said merchant server is authorized to release said item of merchandise or said service to a user associated with said smart card.

9. The cable television payment system according to claim 8, wherein said cable head end server further includes a command emulator for emulating security card commands that are sent to said smart card and for grouping responses to said security card commands into a draw request message to be sent to said payment server, and said payment server includes a response emulator for emulating responses from said smart card that are sent to said security card.

10. The cable television payment system according to claim 8 or 9, wherein said payment server includes a comparator for comparing a smart card signature received from said smart card with an expected signature received from said security card to confirm a transaction.

11. The cable television payment system according to any of claims 8 to 10, wherein said cable head end server includes a comparator for comparing a smart card signature received from said smart card with an expected signature from said security card received via said payment server, whereby a transaction is confirmed from said payment server.

12. A computer-implemented method of transacting a sale of merchandise or services over a network using a cable head end server terminal in association with a smart card, said method comprising:

transmitting over said network a request from said cable head end server terminal to purchase an item or service available from said merchant server;
receiving from said merchant server an amount of a cost of said item or service;
sending a draw request message to a payment server connected to said network so that said draw request may be processed by a security card associated with said payment server;
receiving a debit command from said payment server;
sending said debit command to a set-top box in communication with said cable head end server;
debiting said smart card associated with said set-top box by said amount; and
sending confirmation information to said merchant server, whereby said merchant server is informed that said sale of said item or service is a success and said merchant server may release said item or service to a user associated with said smart card.

13. A method according to claim 12, further comprising:

sending a message indicating that said sale of said item or service is a success from said merchant server to said cable head end server; and
sending said message indicating that said sale of said item or service is a success from said cable head end server to said set-top box.

14. A method according to claim 12 or 13, wherein said network is an internet, wherein said merchant server includes a merchant web site for advertising said merchandise over said internet, wherein each of said cable head end server terminal, said merchant

server and said payment server are at a separate location and said recited steps of said method occur over said internet.

15. A method according to any of claims 12 to 14, further comprising:

emulating security card commands that are sent to said smart card associated with said cable head end server terminal; and
grouping responses to said security card commands into said draw request message so that said responses may be sent as a group to said payment server to reduce network traffic between said payment server and said cable head end server terminal.

16. A method according to any of claims 12 to 15, wherein said confirmation information includes an encrypted confirmation message unreadable by said cable head end server terminal, said method further comprising:

receiving said encrypted confirmation message from said payment server.

17. A method according to any of claims 12 to 16, wherein said confirmation information includes a confirmation message, said method further comprising:

receiving an expected smart card signature from said security card via said payment server;
receiving an actual smart card signature from said smart card;
comparing said actual smart card signature received from said smart card with said expected smart card signature from said security card; and
generating said confirmation message for transmission to said merchant server, whereby message traffic between said payment server and said cable head end server terminal, and between said payment server and said security card is reduced.

18. A method according to any of claims 12 to 16, further comprising:

receiving an encrypted smart card signature from said security card via said payment server that is unreadable by said cable head end server terminal;
receiving a raw smart card signature from said smart card; and
transmitting to said merchant server as said confirmation information said encrypted smart card signature and said raw smart card signa-

ture for comparison by said merchant server, whereby message traffic between said payment server and said cable head end server terminal, and between said payment server and said security card is reduced.

19. A method according to any of claims 12 to 18, further comprising:

receiving a key from said merchant server for encrypting said draw request message to be sent to said payment server;
receiving an encrypted version of said key that is unreadable by said cable head end server terminal, said key being encrypted using a shared key that is known to said payment server and to said merchant server; and
sending said encrypted version of said key to said payment server without sending said key in the clear to said payment server, whereby said payment server may decrypt and obtain said key to decrypt said draw request message.

20. A method according to any of claims 12 to 19, further comprising:

receiving a security card signature for validating said security card to said smart card, said security card signature being received in the same message from said payment server as said debit command; and
receiving an expected smart card signature for comparison to an actual smart card signature, said expected smart card signature being received in the same message from said payment server as said debit command, whereby message traffic between said payment server and said cable head end server terminal, and between said payment server and said security card is reduced.

21. A computer-implemented method of managing a transaction between a cable head end server terminal and a merchant server connected over a network, said transaction being managed by a payment server also connected to said network, said method comprising:

receiving a draw request over said network, said draw request including an amount indicative of a cost of an item or service available from said merchant server, a transaction identifier uniquely identifying the purchase of said item or service, and a merchant identifier uniquely identifying said merchant server to said payment server;
sending said draw request to a security card associated with said payment server so that said

draw request may be processed by said security card;
receiving a debit command from said security card;
sending said debit command from said payment server to said cable head end server;
sending said debit command from said cable head end server to a set-top box in communication with said cable head end server so that a smart card associated with said set-top box may be debited by said amount; and
a confirmation step for performing the function of confirming said purchase of said item or service to said merchant server, whereby said merchant server is informed that said purchase of said item is a success and said merchant server may release said item or service to a user associated with said smart card.

22. A method according to claim 21, wherein said network is an internet, wherein said merchant server includes a merchant web site for advertising said item or service over said internet, wherein each of said cable head end server terminal, said merchant server and said payment server are at a separate location and said recited steps of said method occur over said internet.

23. A method according to claim 21 or 22, wherein said smart card and said security card are both also suitable for use in an integrated service payment terminal, said method further comprising:

sending transaction information regarding said sale of said item or service to a clearing and administration system for reconciling said sale.

24. A method according to any of claims 21 to 23, further comprising:

receiving as part of said draw request responses from said smart card to security card commands that have been emulated by said cable head end server terminal; and
emulating said smart card responses in an interaction with said security card to receive responses from said security card, whereby network traffic between said payment server and said cable head end server terminal is reduced.

25. A method according to any of claims 12 to 24, wherein said confirmation step includes the substeps of:

receiving a signature from said smart card associated with said cable head end server terminal;
sending said signature to said security card;

receiving a transaction OK message from said security card; and

sending a confirmation message destined for said merchant server.

26. A method according to any of claims 12 to 24, wherein said confirmation step includes the sub-steps of:

receiving a signature from said smart card associated with said cable head end server terminal;

comparing said received signature with an expected signature received from said security card; and

sending a confirmation message destined for said merchant server, whereby message traffic between said security card and said payment server is reduced.

27. A method according to any of claims 12 to 24, wherein said confirmation step includes the sub-steps of:

receiving an expected signature of said smart card from said security card; and

sending said expected signature to said cable head end server terminal so that said cable head end server terminal may compare said expected signature to an actual signature of said smart card, whereby message traffic between said security card and said payment server, and between said cable head end server terminal and said payment server is reduced.

28. A method according to any of claims 12 to 24, wherein said confirmation step includes the sub-steps of:

receiving an expected signature of said smart card from said security card;

encrypting said expected signature so as to be unreadable by said cable head end server terminal; and

sending said encrypted expected signature to said cable head end server terminal for resending to said merchant server so that said merchant server may compare said expected signature to an actual signature of said smart card, whereby message traffic between said security card and said payment server, and between said cable head end server terminal and said payment server is reduced.

29. A method according to any of claims 12 to 28, further comprising:

sending a security card signature for validating

said security card, said security card signature being sent in the same message destined to said cable head end server terminal as said debit command; and

sending an expected smart card signature for comparison to an actual smart card signature, said expected smart card signature being sent in the same message destined to said cable head end server terminal as said debit command, whereby message traffic between said payment server and said cable head end server terminal, and between said payment server and said security card is reduced.

30. A method according to any of claims 12 to 29, wherein said steps of receiving a draw request, sending said debit command and said confirmation step are routed through said cable head end server terminal to reduce network traffic.

FIG. 1
(PRIOR ART)

FIG. 2

FIG. 3

FIG. 4

FIG. 5

FIG. 6

Smart Card
(SVC) 5

Debit Success, encrypted SVC
Signature, raw SVC
Signature

235

360

Merchant
Server

366

Compare SVC
Signatures 208

Cable head end
Server 204

306

304

302

316b
Debit,
Security Signature,
Encrypted SVC
Signature

Debit,
Security Signature,
SVC Signature

Terminal 214

Security
Card 218

Draw
Request

200c

310

Merchant
Payment
Server

206

314a

Draw
Request

312

FIG. 7

34

FIG. 8

FIG. 9

```
                    ┌─────────────────────────────┐
                    │           BEGIN             │
                    │ INTERNET PURCHASE BY USER WITH │
                    │         SMART CARD          │
                    └─────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      502
        │   USER ACQUIRES AND ADDS VALUE TO A SMART CARD │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      504
        │   USER ACCESSES MERCHANT SERVER WEB SITE OVER  │
        │            INTERNET VIA SET-TOP BOX            │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      506
        │   USER INSERTS CARD IN CARD READER AT SET-TOP BOX │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      508
        │ USER SELECTS GOODS AND/OR SERVICES FOR PURCHASE │
        │             FROM MERCHANT WEB SITE             │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      510
        │   USER RECEIVES TOTAL SALE AND SELECTS "PURCHASE │
        │               WITH SMART CARD"                 │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      512
        │ CABLE TELEVISION PAYMENT SYSTEM PROCESSES ORDER │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      514
        │   USER SMART CARD DEBITED AND USER RECEIVES    │
        │   "DEBITED" MESSAGE FROM MERCHANT SERVER       │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      516
        │ USER RECEIVES CONFIRMATION OF SALE FROM MERCHANT │
        │ SERVER AND RECEIVES DOWNLOADED INFORMATION OR  │
        │ RECEIPT FOR GOODS AND/OR SERVICES TO BE RENDERED │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
        ┌───────────────────────────────────────────────┐      518
        │ MERCHANT RECEIVES PAYMENT TO BANK ACCOUNT BY   │
        │   WAY OF INFORMATION FROM MERCHANT PAYMENT     │
        │                 SERVER                         │
        └───────────────────────────────────────────────┘
                                 │
                                 ▼
                          ┌─────────────┐
                          │     END     │
                          └─────────────┘
```

FIG. 10A

**BEGIN
PROCESS USER PURCHASE BY SYSTEM
(see FIGURE 5)**

USER SELECTS GOODS AND/OR SERVICES FROM MERCHANT SITE AND
REQUESTS PURCHASE WITH SMART CARD — 602

MERCHANT SERVER RECEIVES REQUESTS FOR
SMART CARD TRANSACTION — 604

MERCHANT SERVER SENDS PAGE OF INFORMATION TO CABLE HEAD END
INCLUDING: TOTAL COST, CURRENCY, IP ADDRESS OF PAYMENT SERVER,
TRANSACTION IDENTIFIER, MERCHANT IDENTIFIER — 606

CABLE HEAD END SERVER SENDS THE PAGE TO SET-TOP BOX — 607

SET-TOP BOX INTERACTS WITH SMART CARD READER AND SENDS
INFORMATION TO CABLE HEAD END SERVER — 608

CABLE HEAD END BUILDS DRAW REQUEST MESSAGE — 609

CABLE HEAD END ACCESSES PAYMENT SERVER USING IP ADDRESS — 610

CABLE HEAD END SERVER SENDS DRAW REQUEST MESSAGE PLUS ADDITIONAL
INFORMATION RECEIVED FROM MERCHANT SERVER TO PAYMENT SERVER — 612

PAYMENT SERVER PROCESSES DRAW REQUEST IN
CONJUNCTION WITH SECURITY CARD
(see Figure 11D) — 614

PAYMENT SERVER SENDS "DEBIT" COMMAND WITH SECURITY CARD
SIGNATURE TO CABLE HEAD END SERVER IN ORDER FOR SMART CARD TO
DEBIT ITSELF — 616

CABLE HEAD END SENDS DEBIT COMMAND WITH SECURITY
CARD SIGNATURE TO SET-TOP BOX — 617

(A)

FIG. 10B (A)

SET-TOP BOX PASSES "DEBIT" COMMAND TO CARD WHICH DEBITS
ITSELF BY TOTAL SALE AND GENERATES CARD SIGNATURE — 618

CARD SENDS "DEBIT SUCCESS" AND CARD SIGNATURE TO SET-TOP BOX — 620

SET-TOP BOX SENDS "DEBIT SUCCESS" AND CARD SIGNATURE TO CABLE HEAD
END SERVER — 621

CABLE HEAD END SERVER SENDS "DEBIT SUCCESS"
MESSAGE AND CARD SIGNATURE TO PAYMENT SERVER — 622

PAYMENT SERVER DIRECTS RECEIVED RESPONSE TO
SECURITY CARD IN TERMINAL — 624

SECURITY CARD PROCESSES RESPONSE AND VERIFIES CARD SIGNATURE — 626

SECURITY CARD SENDS "CONFIRMATION" MESSAGE TO PAYMENT SERVER — 628

TERMINAL UPDATES ITS DATA STORE WITH CARD NUMBER, TRANSACTION
COUNT, TOTAL SALE, RESPONSE FROM CARD, TRANSACTION NUMBERS FROM
STORED-VALUE CARD AND SECURITY CARD, ETC. — 630

PAYMENT SERVER UPDATES ITS DATABASE WITH LOG OF TRANSACTIONS TO
DATE, MERCHANT IDENTIFIER, ETC. — 632

(B)

```
                              ( B )
                                │
                                ▼
        ┌──────────────────────────────────────────────┐    634
        │ PAYMENT SERVER SENDS "CONFIRMATION" MESSAGE TO │ ╱
        │      CABLE HEAD END IN ENCRYPTED FORM          │
        └──────────────────────────────────────────────┘
                                │
                                ▼
        ┌──────────────────────────────────────────────┐    636
        │ CABLE HEAD END PASSES "CONFIRMATION" MESSAGE TO│ ╱
        │               MERCHANT SERVER                  │
        └──────────────────────────────────────────────┘
                                │
                                ▼
        ┌──────────────────────────────────────────────┐    638
        │       MERCHANT SERVER REGISTERS "CONFIRMATION" │ ╱
        │               MESSAGE                          │
        └──────────────────────────────────────────────┘
                                │
                                ▼
        ┌──────────────────────────────────────────────┐    640
        │ MERCHANT SERVER DELIVERS INFORMATION TO CABLE  │ ╱
        │ HEAD END AND/OR PROVIDES RECEIPT FOR GOODS     │
        │      AND/OR SERVICES TO BE RENDERED            │
        └──────────────────────────────────────────────┘
                                │
                                ▼
        ┌──────────────────────────────────────────────┐    641
        │ CABLE HEAD END SENDS THE INFORMATION TO THE SET│ ╱
        │                  TOP BOX                       │
        └──────────────────────────────────────────────┘
                                │
                                ▼
                            (  END  )
```

FIG. 10C

FIG. 10D BEGIN STEP 614

DRAW REQUEST EDITED SYNTACTICALLY AND LOGGED — 680

DRAW REQUEST PASSED TO TERMINAL INTERFACE MODULE — 682

TERMINAL INTERFACE BUILDS TERMINAL SPECIFIC MESSAGE BASED UPON DRAW REQUEST AND TYPE OF TERMINAL — 684

DRAW REQUEST SENT TO TERMINAL VIA TERMINAL CONCENTRATOR — 686

TERMINAL PARSES DRAW REQUEST INTO COMPONENTS AND PROCESSES EACH COMPONENT IN TURN — 688

TERMINAL REACHES "DRAW AMOUNT" STATE — 690

SECURITY CARD IN TERMINAL GENERATES SECURITY CARD SIGNATURE AND "DEBIT" COMMAND — 692

TERMINAL SENDS SECURITY CARD SIGNATURE AND "DEBIT" COMMAND TO PAYMENT SERVER — 694

END STEP 614

BEGIN ALTERNATIVE
COMPARISON OF SMART CARD SIGNATURE TO
ALLOW SECURITY CARD TO RELEASE EARLIER
(see FIGURE 6)

PERFORM STEPS 602-612 AND STEPS 680-690 — 702

SECURITY CARD IN TERMINAL GENERATES SECURITY CARD — 704
SIGNATURE, "DEBIT" COMMAND AND EXPECTED
SMART CARD SIGNATURE

TERMINAL SENDS SECURITY CARD SIGNATURE, "DEBIT" COMMAND — 706
AND EXPECTED SMART CARD SIGNATURE TO PAYMENT SERVER

PERFORM STEPS 616-622 — 708

PAYMENT SERVER CODE MODULE PROCESSES RESPONSE, — 710
VERIFIES CARD SIGNATURE BY COMPARING RECEIVED CARD
SIGNATURE WITH EXPECTED SMART CARD SIGNATURE
RECEIVED EARLIER FROM SECURITY CARD

PERFORM STEPS 632-640 — 712

FIG. 11   END

BEGIN ALTERNATIVE
COMPARISON OF SMART CARD SIGNATURE
BY HEAD END SERVER
(see FIGURE 7)

722
PERFORM STEPS 602-612 AND STEPS 680-690

724
SECURITY CARD IN TERMINAL GENERATES SECURITY CARD SIGNATURE,
"DEBIT" COMMAND AND EXPECTED SMART CARD SIGNATURE

726
TERMINAL SENDS SECURITY CARD SIGNATURE, "DEBIT" COMMAND AND
EXPECTED SMART CARD SIGNATURE TO PAYMENT SERVER

728
PAYMENT SERVER SENDS "DEBIT" COMMAND, SECURITY CARD SIGNATURE AND
EXPECTED SMART CARD SIGNATURE TO CABLE HEAD END SERVER

730
PERFORM STEPS 618, 620

732
CABLE HEAD END CODE MODULE PROCESSES RESPONSE, VERIFIES CARD
SIGNATURE BY COMPARING RECEIVED CARD SIGNATURE FROM SMART CARD
WITH EXPECTED SIGNATURE RECEIVED EARLIER FROM SECURITY CARD VIA
PAYMENT SERVER

734
CABLE HEAD END CODE MODULE GENERATES CONFIRMATION
MESSAGE FOR MERCHANT SERVER

736
PERFORM STEPS 636-640

FIG. 12

END

BEGIN ALTERNATIVE
COMPARISON OF SMART CARD SIGNATURE
BY MERCHANT SERVER
(see FIGURE 8)

PERFORM STEPS 602-612 AND STEPS 680-690 — 742

SECURITY CARD IN TERMINAL GENERATES SECURITY CARD SIGNATURE, "DEBIT" COMMAND AND EXPECTED SMART CARD SIGNATURE — 744

TERMINAL SENDS SECURITY CARD SIGNATURE, "DEBIT" COMMAND AND EXPECTED SMART CARD SIGNATURE TO PAYMENT SERVER — 746

PAYMENT SERVER SENDS "DEBIT" COMMAND, SECURITY CARD SIGNATURE AND ENCRYPTED SMART CARD SIGNATURE TO CABLE HEAD END SERVER — 748

PERFORM STEPS 618, 620 — 750

CABLE HEAD END SERVER SENDS DEBIT SUCCESS MESSAGE, RAW CARD SIGNATURE RECEIVED FROM SMART CARD AND ENCRYPTED CARD SIGNATURE RECEIVED FROM PAYMENT SERVER TO MERCHANT SERVER — 752

MERCHANT SERVER PROCESSES DEBIT SUCCESS MESSAGE AND DECRYPTS ENCRYPTED CARD SVC SIGNATURE IN ORDER TO COMPARE RAW CARD SIGNATURE TO CARD SIGNATURE FROM SECURITY CARD — 754

MERCHANT SERVER GENERATES CONFIRMATION MESSAGE — 756

PERFORM STEPS 638, 640 — 758

FIG. 13   END

BEGIN IMPLEMENTATION
OF ADDED SECURITY LAYER TO EMBODIMENTS
OF THE INVENTION
(see FIGURE 9)

PAYMENT SERVER AND MERCHANT SERVER SHARE A
UNIQUE DES ENCRYPTION KEY — 802

CABLE HEAD END SERVER AND MERCHANT SERVER
ENGAGE IN PROTECTED SSL SESSION — 804

MERCHANT SERVER DERIVES A KEY FROM THE DES KEY
USING INFORMATION UNIQUE TO THE TRANSACTION SUCH
AS MERCHANT IDENTIFIER, TRANSACTION IDENTIFIER, ETC. — 806

MERCHANT SERVER DOWNLOADS HTML PAGE TO CABLE
HEAD END INCLUDING A TRANSACTION SESSION KEY (TSK)
AND THE TSK ENCRYPTED WITH DERIVED KEY (ETSK) — 808

PAGE SENT TO SET-TOP BOX — 809

CABLE HEAD END SERVER SENDS DRAW REQUEST
ENCRYPTED WITH TSK TO PAYMENT SERVER ALONG WITH
ETSK — 810

PAYMENT SERVER DECRYPTS ETSK WITH SHARED DES
KEY TO PRODUCE TSK; DECRYPTS DRAW REQUEST WITH
TSK IN ORDER TO PROCESS DRAW REQUEST, AND
ENCRYPTS "DEBIT" COMMAND WITH TSK — 812

A

FIG. 14A

A

PAYMENT SERVER SENDS ENCRYPTED "DEBIT" COMMAND TO CABLE HEAD END SERVER — 814

CABLE HEAD END SERVER DECRYPTS "DEBIT" COMMAND AND SENDS TO SET-TOP BOX — 816

SET-TOP BOX PROCESSES IN CONJUNCTION WITH STORED-VALUE CARD — 817

SET-TOP BOX SENDS "DEBIT RESPONSE" MESSAGE ENCRYPTED WITH TSK TO CABLE HEAD END SERVER — 818

CABLE HEAD END SENDS THE MESSAGE TO PAYMENT SERVER — 819

PAYMENT SERVER AND SECURITY CARD PROCESS "DEBIT RESPONSE" MESSAGE AND SEND "DEBIT RESULT C" MESSAGE ENCRYPTED WITH TSK AND "DEBIT RESULT M" MESSAGE ENCRYPTED WITH DERIVED KEY TO CABLE HEAD END — 820

CABLE HEAD END DECRYPTS "DEBIT RESULT C" MESSAGE AND PASSES "DEBIT RESULT M" MESSAGE ON TO MERCHANT SERVER — 822

MERCHANT SERVER DECRYPTS "DEBIT RESULT M" MESSAGE USING DERIVED KEY FROM DES KEY AND PROCESSES RESULT — 824

MERCHANT SERVER SENDS PROPRIETARY MESSAGE INDICATING SUCCESS OR FAILURE TO CABLE HEAD END — 825

CABLE HEAD END SENDS TO SET-TOP BOX — 826

END

FIG. 14B

FIG. 15A

BEGIN
LOAD VALUE ONTO SMART CARD

870

USER SELECTS LOAD VALUE OPTION FROM BANK INTERNET SITE VIA CABLE HEAD END SERVER — 871

SET-TOP BOX READS CARD BALANCE, ETC., AND SENDS TO BANK SERVER VIA CABLE HEAD END SERVER — 872

BANK SERVER DETERMINES MAXIMUM LOAD VALUE AND CHECKS FUNDS IN USER'S ACCOUNT — 873

BANK SERVER SENDS PAGE OF INFORMATION TO CABLE HEAD END INCLUDING: MAXIMUM LOAD VALUE, CURRENCY, IP ADDRESS OF LOAD SERVER, TRANSACTION IDENTIFIER, BANK IDENTIFIER, SESSION KEY — 874

CABLE HEAD END SENDS THE PAGE TO SET-TOP BOX — 888

USER ENTERS AMOUNT TO BE LOADED AND USER ACCOUNT DEBITED BY LOAD VALUE — 875

SET-TOP BOX INTERACTS WITH SMART CARD IN READER AND SENDS RESULT OF INTERACTION TO CABLE HEAD END SERVER — 876

CABLE HEAD END BUILDS LOAD REQUEST MESSAGE — 889

CABLE HEAD END SERVER ACCESSES LOAD SERVER USING IP ADDRESS — 877

CABLE HEAD END SERVER SENDS LOAD REQUEST MESSAGE TO LOAD SERVER — 878

LOAD SERVER PROCESSES LOAD REQUEST — 879

A

FIG. 15B

(A) 870

LOAD SERVER SENDS "LOAD" COMMAND WITH SECURITY MODULE SIGNATURE TO CABLE HEAD END	880
CABLE HEAD END SENDS "LOAD" COMMAND WITH SECURITY MODULE SIGNATURE TO SET-TOP BOX	893
SET-TOP BOX PASSES "LOAD" COMMAND TO CARD WHICH LOADS ITSELF BY LOAD VALUE AND GENERATES CARD SIGNATURE	881
CARD SENDS "LOAD SUCCESS" AND CARD SIGNATURE TO SET-TOP BOX	882
SET-TOP BOX SENDS THE "LOAD SUCCESS" AND CARD SIGNATURE TO CABLE HEAD END SERVER	894
CABLE HEAD END SENDS "LOAD SUCCESS" MESSAGE AND CARD SIGNATURE TO LOAD SERVER	883
LOAD SERVER PROCESSES AND DIRECTS RECEIVED RESPONSE TO HOST SECURITY MODULE	884
HOST SECURITY MODULE VERIFIES CARD SIGNATURE	885
HOST SECURITY MODULE SENDS "CONFIRMATION" TO LOAD SERVER	886
LOAD SERVER UPDATES ITS DATABASE	887

(B)

FIG. 15C   (B)   870

```
┌─────────────────────────────────────────────┐
│ LOAD SERVER SENDS "CONFIRMATION" MESSAGE TO  │  890
│           CABLE HEAD END SERVER              │
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│  CABLE HEAD END SERVER PASSES "CONFIRMATION" │  891
│           MESSAGE TO BANK SERVER             │
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│  BANK SERVER REGISTERS "CONFIRMATION" MESSAGE│  892
└─────────────────────────────────────────────┘
```

(END)

```
┌─────────────────────────────────────────────┐
│  LOAD REQUEST EDITED SYNTACTICALLY AND LOGGED│  895
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│ LOAD SERVER PARSES LOAD REQUEST INTO COMPONENTS AND │  896
│       PROCESSES EACH COMPONENT IN TURN       │
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│ LOAD REQUEST COMPONENTS PASSED TO HOST SECURITY │  897
│                   MODULE                     │
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│  HOST SECURITY MODULE VERIFIES CARD SIGNATURE,  │  898
│ GENERATES SECURITY CARD SIGNATURE, "LOAD" COMMAND │
└─────────────────────────────────────────────┘
```

```
┌─────────────────────────────────────────────┐
│  HOST SECURITY MODULE SENDS MERCHANT CARD    │  899
│ SIGNATURE AND "LOAD" COMMAND TO LOAD SERVER  │
└─────────────────────────────────────────────┘
```

879   FIG. 15D

EP 1 263 230 A1

FIG. 16A BEGIN CABLE PROVIDER PURCHASE BY USER WITH STORED VALUE CARD (see FIGURE 4)

USER ACQUIRES AND ADDS VALUE TO A SMART CARD — 1102

USER ACCESSES CABLE HEAD END SERVER — 1104

USER INSERTS CARD IN CARD READER AT SET-TOP BOX — 1106

USER SELECTS GOODS AND/OR SERVICES FOR PURCHASE FROM CABLE HEAD END — 1108

USER RECEIVES TOTAL SALE AND SELECTS "PURCHASE WITH SMART CARD" — 1110

CABLE TELEVISION PAYMENT AND LOAD SYSTEM PROCESSES ORDER — 1112

USER SMART CARD DEBITED AND USER RECEIVES "DEBITED" MESSAGE FROM CABLE HEAD END MODULE — 1114

USER RECEIVES CONFIRMATION OF SALE FROM CABLE HEAD END SERVER AND RECEIVES DOWNLOADED INFORMATION OR RECEIPT FOR GOODS AND/ OR SERVICES TO BE RENDERED — 1116

CABLE PROVIDER RECEIVES PAYMENT TO BANK ACCOUNT BY WAY OF INFORMATION FROM CABLE HEAD END PAYMENT SERVER — 1118

END

BEGIN LOAD VALUE ONTO SMART CARD — 1070

USER SELECTS LOAD VALUE OPTION FROM A LOADING SITE VIA CABLE HEAD END — 1071

SET-TOP BOX READS CARD BALANCE, ETC., AND SENDS TO LOAD SERVER — 1072

LOAD SERVER DETERMINES MAXIMUM LOAD VALUE AND CHECKS FUNDS IN PRE-FUNDED ACCOUNT — 1073

LOAD SERVER SENDS PAGE OF INFORMATION TO CABLE HEAD END INCLUDING: MAXIMUM LOAD VALUE, CURRENCY, IP ADDRESS OF LOAD SERVER, TRANSACTION IDENTIFIER, BANK IDENTIFIER, SESSION KEY — 1074

SEND THE PAGE TO THE SET-TOP BOX AND REQUEST AMOUNT TO BE LOADED FROM USER — 1088

PRE-FUNDED ACCOUNT DEBITED BY LOAD VALUE — 1075

SET-TOP BOX INTERACTS WITH SMART CARD IN READER AND SENDS RESULT OF INTERACTION TO CABLE HEAD END — 1076

CABLE HEAD END BUILDS LOAD REQUEST MESSAGE — 1089

CABLE HEAD END ACCESSES LOAD SERVER USING IP ADDRESS — 1077

CABLE HEAD END SENDS LOAD REQUEST MESSAGE TO LOAD SERVER — 1078

LOAD SERVER PROCESSES LOAD REQUEST — 1079

A

FIG. 17A

FIG. 17B   (A)   1070

LOAD SERVER SENDS "LOAD" COMMAND WITH SECURITY MODULE SIGNATURE TO CABLE HEAD END SERVER   1080

CABLE HEAD END SENDS THE "LOAD" COMMAND WITH SECURITY MODULE SIGNATURE TO SET-TOP BOX   1093

SET-TOP BOX PASSES "LOAD" COMMAND TO CARD WHICH LOADS ITSELF BY LOAD VALUE AND GENERATES CARD SIGNATURE   1081

CARD SENDS "LOAD SUCCESS" AND CARD SIGNATURE TO SET-TOP BOX   1082

SET-TOP BOX SENDS THE "LOAD SUCCESS" AND CARD SIGNATURE TO CABLE HEAD END SERVER   1094

CABLE HEAD END SENDS "LOAD SUCCESS" MESSAGE AND CARD SIGNATURE TO LOAD SERVER   1083

LOAD SERVER PROCESSES AND DIRECTS RECEIVED RESPONSE TO HOST SECURITY MODULE   1084

HOST SECURITY MODULE VERIFIES CARD SIGNATURE   1085

HOST SECURITY MODULE SENDS "CONFIRMATION" TO LOAD SERVER   1086

LOAD SERVER UPDATES ITS DATABASE   1087

(B)

FIG. 17C

(B)

─1070

| LOAD SERVER SENDS "CONFIRMATION" MESSAGE TO CABLE HEAD END SERVER | ─1090 |

| CABLE HEAD END PASSES "CONFIRMATION" MESSAGE TO USER | ─1091 |

(END)

| LOAD REQUEST EDITED SYNTACTICALLY AND LOGGED | ─1095 |

| LOAD SERVER PARSES LOAD REQUEST INTO COMPONENTS AND PROCESSES EACH COMPONENT IN TURN | ─1096 |

| LOAD REQUEST COMPONENTS PASSED TO SECURITY MODULE | ─1097 |

| HOST SECURITY MODULE VERIFIES CARD SIGNATURE, GENERATES SECURITY CARD SIGNATURE, "LOAD" COMMAND | ─1098 |

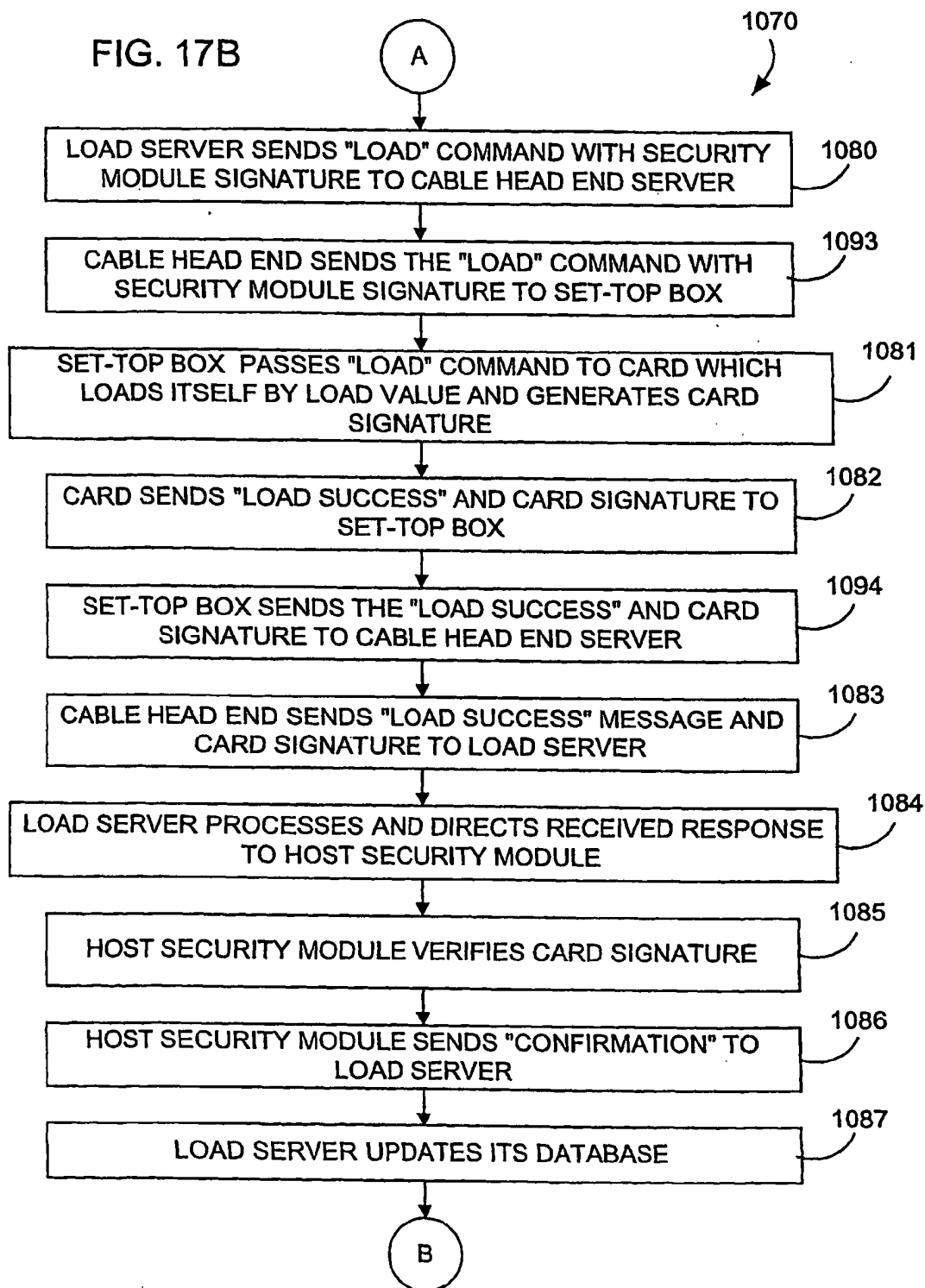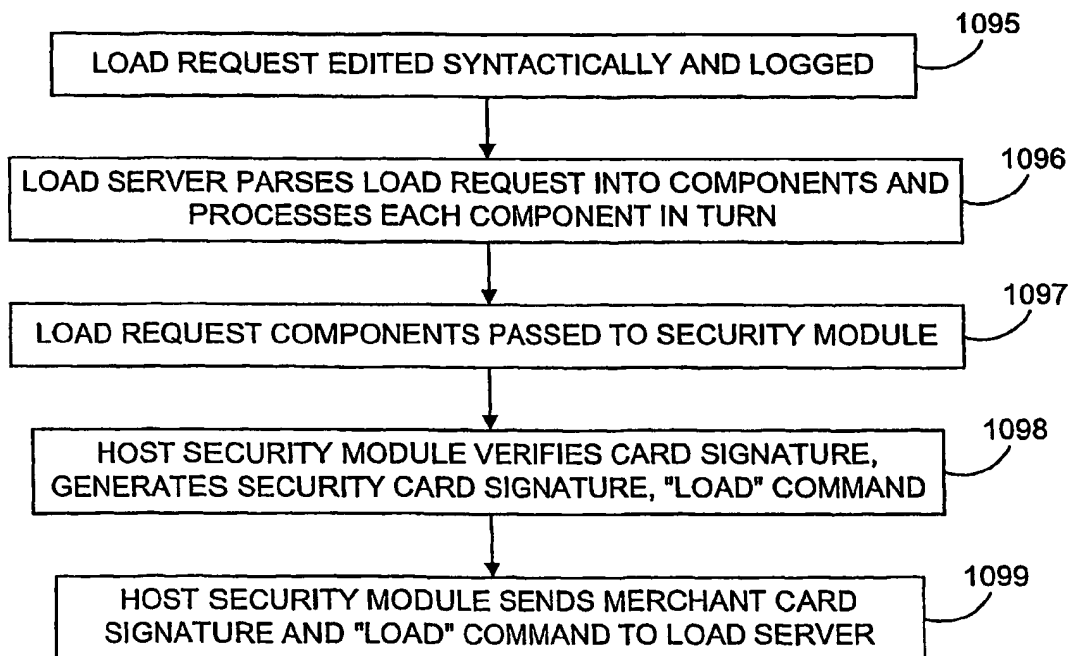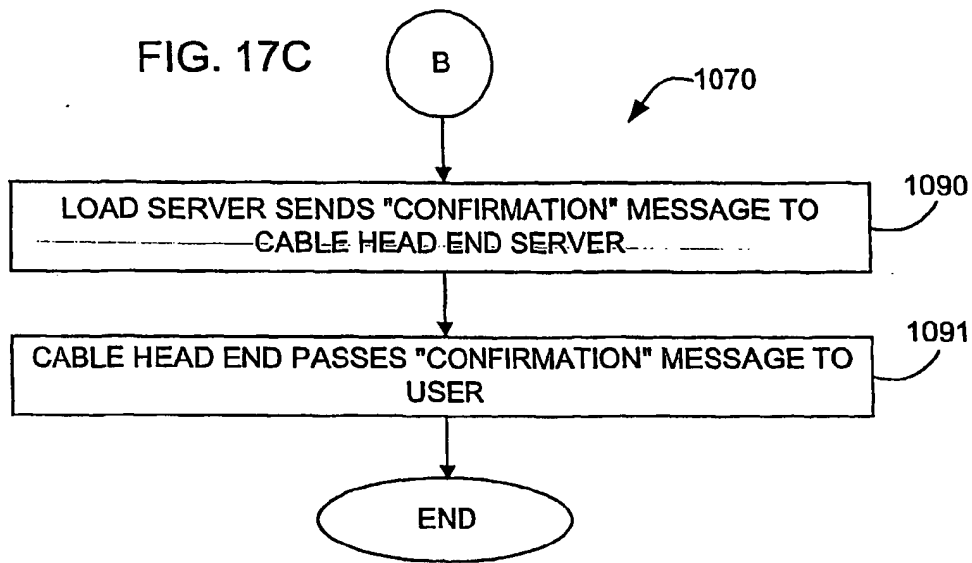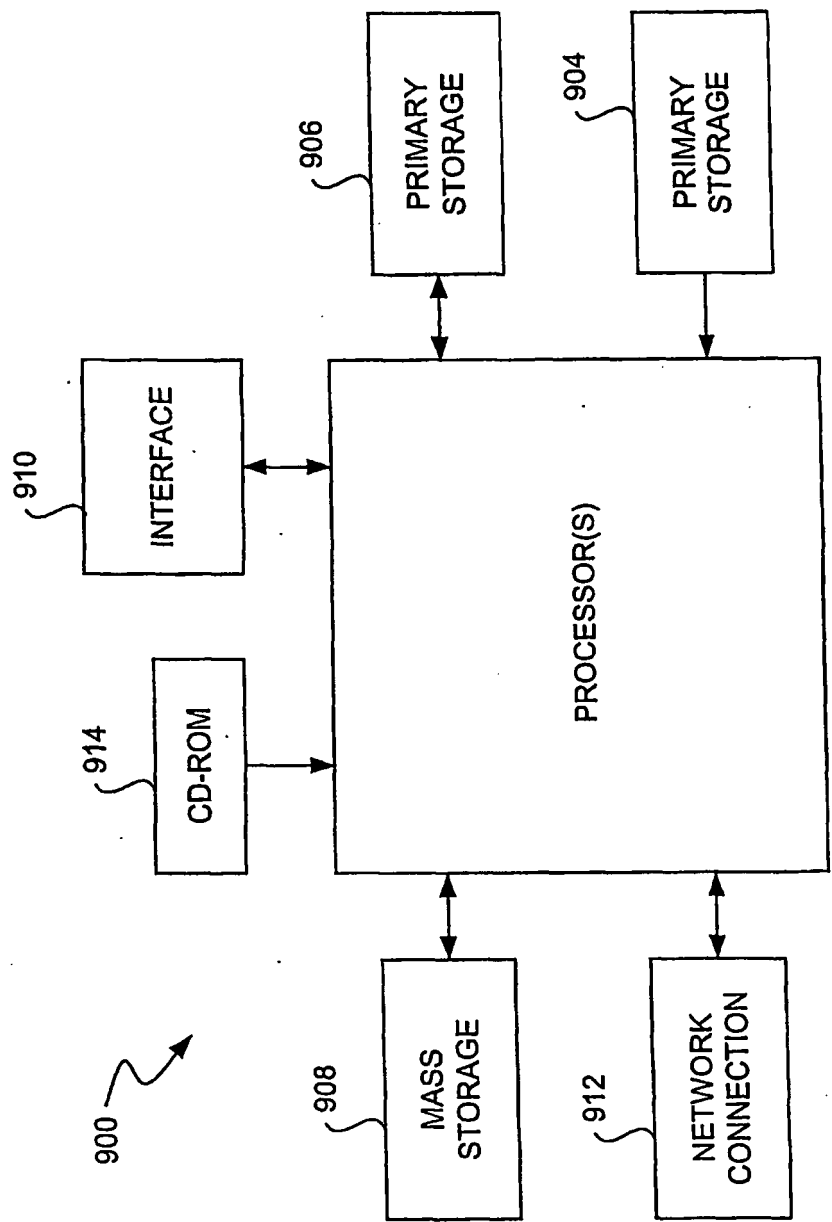| HOST SECURITY MODULE SENDS MERCHANT CARD SIGNATURE AND "LOAD" COMMAND TO LOAD SERVER | ─1099 |

1079 ─

FIG. 17D

FIG. 18

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 02 01 7779

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 5 790 677 A (FOX ET AL) 4 August 1998 (1998-08-04)	1,8	H04N7/173 G07F19/00
Y	* column 6, line 30 - line 35 *	2,3	
A	* column 7, line 59 - line 67 * * column 19, line 38 - line 43 * * column 28, line 57 - line 65; figure 23 * * column 29, line 7 - column 10; figure 23 * * column 29, line 36 - line 67; figure 23 *	12,21	

Y	US 5 883 810 A (FRANKLIN ET AL) 16 March 1999 (1999-03-16)	2,3	
A	* column 1, line 19 - line 25 * * column 3, line 48 - column 4, line 5 *	1,8,12, 21	

A	US 5 392 353 A (MORALES) 21 February 1995 (1995-02-21) * abstract *	1,8,12, 21	
	---		**TECHNICAL FIELDS SEARCHED** (Int.Cl.7)
A	WO 98 49658 A (VISA) 5 November 1998 (1998-11-05) * abstract *	1,8,12, 21	H04N G07F

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	10 October 2002	Dockhorn, H

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.

EP 02 01 7779

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

10-10-2002

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5790677	A	04-08-1998	NONE		
US 5883810	A	16-03-1999	NONE		
US 5392353	A	21-02-1995	US	5663757 A	02-09-1997
			US	5101267 A	31-03-1992
			AU	634242 B2	18-02-1993
			AU	5693790 A	20-12-1990
			CA	2018539 A1	13-12-1990
			CN	1048136 A ,B	26-12-1990
			CS	9002906 A2	12-11-1991
			EP	0402809 A2	19-12-1990
			IE	902108 A1	02-01-1991
			IL	94519 A	08-07-1993
			JP	3064286 A	19-03-1991
			NO	902571 A ,B,	14-12-1990
			NZ	233837 A	28-04-1992
			PT	94361 A	31-12-1991
			US	5257099 A	26-10-1993
			US	5223923 A	29-06-1993
WO 9849658	A	05-11-1998	US	6282522 B1	28-08-2001
			AU	7272698 A	24-11-1998
			EP	1003139 A2	24-05-2000
			EP	1023705 A1	02-08-2000
			WO	9849658 A1	05-11-1998
			US	6105008 A	15-08-2000